

Linear Game Automata: Decidable Hierarchy Problems for Stripped-Down Alternating Tree Automata

Jacques Duparc¹, Alessandro Facchini^{1,2*}, and Filip Murlak^{3**}

¹ University of Lausanne

² University of Bordeaux 1

³ University of Edinburgh

{jacques.duparc, alessandro.facchini}@unil.ch,
fmurlak@inf.ed.ac.uk

Abstract. For deterministic tree automata, classical hierarchies, like Mostowski-Rabin (or index) hierarchy, Borel hierarchy, or Wadge hierarchy, are known to be decidable. However, when it comes to non-deterministic tree automata, none of these hierarchies is even close to be understood. Here we make an attempt in paving the way towards a clear understanding of tree automata. We concentrate on the class of linear game automata (LGA), and prove within this new context, that all corresponding hierarchies mentioned above—Mostowski-Rabin, Borel, and Wadge—are decidable. The class LGA is obtained by taking linear tree automata with alternation restricted to the choice of path in the input tree. Despite their simplicity, LGA recognize sets of arbitrary high Borel rank. The actual richness of LGA is revealed by the height of their Wadge hierarchy: $(\omega^\omega)^\omega$.

1 Introduction

Mostowski–Rabin hierarchy, Borel hierarchy, and Wadge hierarchy are the most common measures of complexity of recognizable ω -languages.

The first one, also known as the index hierarchy, orders languages according to the nesting of positive and negative conditions checked by the recognizing automaton. It has two main versions: weak, which relies on finitary conditions (e.g., “ a does occur”); and strong, referring to infinitary conditions (e.g., “ b occurs infinitely often”). Mostowski–Rabin hierarchy is believed to reflect the inherent combinatorial complexity of all possible automata recognizing the language, and therefore has attracted a lot of attention encouraged by the expectations of the verification community [3,4,10,17,18,19,20].

The classical Borel hierarchy is based on the nesting of countable unions and negations in the set theoretic definition of the language, starting from the simplest (open) sets. It drew attention of automata theorists as early as 1960s [12], and has continued

* Research supported by a grant from the Swiss National Science Foundation, n. 100011-116508: Project “Topological Complexity, Games, Logic and Automata”.

** On leave from the University of Warsaw; partially supported by the Polish government grant no. N206 008 32/0810.

to inspire research efforts ever since, mainly because of its intimate relations with the index hierarchy [9,19,22].

The Wadge hierarchy is an almost ultimate refinement of the Borel hierarchy, defined by the preorder induced on languages by simple (continuous) reductions. One of the amazing possibilities it offers is a precise comparison of different models of computation. What is more powerful: deterministic or weak automata on trees? It is known that there are deterministic languages that are not weakly recognizable and vice versa. How to compare, if not by inclusion? An even more exotic case: deterministic tree languages versus deterministic context free word languages. How to compare trees with words? The Wadge hierarchy makes it possible. The sole heights (huge ordinals) of the Wadge hierarchy restricted to the classes under comparison provide, literally, infinitely more information than other logical techniques [6,8,15,21].

Measuring hardness of recognizable languages of infinite trees is a long standing open problem. Unlike for infinite words — where the understanding is almost complete since Wager’s 1977 paper [24] — for trees the only satisfyingly examined case is that of deterministic automata [14,15,16,19,20]. But the deterministic and non-deterministic case differ immensely in the case of trees. The only results obtained for non-deterministic or alternating automata are strictness theorems for various classes [3,4,13,17], and lower bounds for the heights of the hierarchies [22,7]. To the best of our knowledge, the only nontrivial decidability result for non-deterministic tree automata is the famous Rabin’s theorem, which, despite its universally acknowledged importance, gives rather rudimentary information on the complexity of the language in question.

This paper aims at modifying this situation, even if only very slightly for a start. We propose a class of automata with all three hierarchies decidable —Mostowski–Rabin, Borel, and Wadge— that captures a reasonable amount of non-determinism. The class we advocate, *linear game automata*, is obtained by taking linear automata (a.k.a. very weak automata), which emerged in the verification community, and restricting the alternation to the choice of a path in the input tree. The linear automata capture CTL [11], which is expressive enough for many applications. Linear game automata are weaker, nevertheless they preserve most alternations related to the branching structure. Evidence for the expressivity of LGA comes from their topological properties: they recognize sets of arbitrarily high finite Borel rank, and their Wadge hierarchy has the height $(\omega^\omega)^\omega$, much higher than $(\omega^\omega)^3 + 3$ for deterministic automata.

As we already pointed out, these automata are very weak; so this is just a very first step on the way to the complete understanding of alternating automata on infinite trees. This notwithstanding, computing the Wadge degree of a LGA is much more involved than calculating the Wadge degree of an ω -word automaton and even a deterministic tree automaton. If the shape of these automata seems to bring down the calculation to the decomposition of nested chains, alternation (even if very weak) makes everything much harder to compute, more expressive and complicated. We also believe that the notion of game automata is well suited to take us further. Indeed, the next step is to consider weak game automata, then strong game automata, which inhabitate every level of the (strong) index hierarchy and subsume deterministic languages. The complete understanding of this last class would be an important step towards the understanding of (strong) alternating tree automata.

2 Preliminaries

2.1 Weak automata

Let W be a non empty set. A tree over Σ is a partial function $t : W^* \rightarrow \Sigma$ with a prefix closed domain. A tree is called *full* if $\text{dom}(t) = W^*$, and it is called *binary* if $W = \{0, 1\}$. Let T_Σ denote the set of full binary trees over Σ . In the sequel we only consider full binary trees. Given $v \in \text{dom}(t)$, by $t.v$ we denote the subtree of t rooted in v . We write \bar{d} to denote the other direction: $\bar{0} = 1, \bar{1} = 0$.

A *weak alternating tree automaton* $A = \langle \Sigma, Q, q_I, \delta, \text{rank} \rangle$ consists of a finite input alphabet Σ , a finite set of states Q partitioned into the existential states Q_\exists and the universal states Q_\forall , an initial state q_I , a transition relation $\delta \subseteq Q \times \Sigma \times \{\epsilon, 1, 0\} \times Q$ and a bounded priority function $\text{rank} : Q \rightarrow \omega$. Sometimes we write $q \xrightarrow{\sigma, d} q'$ when $q' \in \delta(q, \sigma, d)$. The acceptance is defined in terms of a (weak) parity game.

A *parity game* is a two-player game given by a graph $\langle V, V_0, V_1, E, \text{rank} \rangle$, where $V = V_0 \cup V_1$ is the set of vertices, $E \subseteq V \times V$ is the edge relation and priority or coloring function $\text{rank} : V \rightarrow \omega$, which is bounded. A vertex v is a successor of a vertex v' if $(v', v) \in E$ (we also write $v \in E(v')$). A play from some vertex v consists on a path $v_0 v_1 v_2 \dots$ of a token along the edges of the graph. If the token is in $v \in V_i$, Player i chooses the next location of the token amount $E(v)$. The winning condition for a parity game is given as follows: if a player cannot make a move, he loses; if the play is infinite, Player 0 wins if and only if the greatest priority occurring infinitely often in the sequence $\text{rank}(v_0)\text{rank}(v_1)\text{rank}(v_2) \dots$ is even. A parity game is called a *weak parity game* if, as a winning condition, we take that player 0 wins (either a finite or an infinite play) if and only if the greatest priority occurring in the play is even.

Consider a weak alternating automaton A and an tree $t \in T_\Sigma$. The automaton A accepts t iff Player 0 has a winning strategy in the *weak parity game* $\mathcal{G}_{A,t}$ defined as:

- $V_0 = \{0, 1\}^* \times Q_\exists, V_1 = \{0, 1\}^* \times Q_\forall$,
- the relation $E = \{((v, p), (vd, q)) : v \in \text{dom}(t), (p, t(v), d, q) \in \delta\}$,
- $\text{rank}((v, q)) = \text{rank}(q)$, for every vertex (v, q) .

Given an automaton A , a path in A is a sequence of states and transitions

$$q_0 \xrightarrow{\sigma_0, d_0} q_1 \xrightarrow{\sigma_1, d_1} q_2 \cdots q_n \xrightarrow{\sigma_n, d_n} q_{n+1}$$

If there is such a path with $q = q_0$ and $q' = q_{n+1}$, we say that q' is *reachable* from q . We say that a path is a *loop* if $q_{n+1} = q_0$. If there is a loop from a state q , we say that this state is looping. If q is looping and $\text{rank}(q)$ is even (resp. odd) we say that the loop in q is positive (resp. negative). Finally, we say that a state p is replicated by q if there is a path $q \xrightarrow{\sigma_0, d_0} q_1 \cdots q_n \xrightarrow{\sigma_n, d_n} p$ and a transition $q \xrightarrow{\sigma_0, d_0} q$.

2.2 Borel classes and Wadge reductions

Consider the space T_Σ equipped with the standard Cantor topology. Recall that the class of Borel sets of a topological space X is the closure of the class of open sets of X by countable unions and complementation. Given X , the initial finite levels of the Borel hierarchy are defined as follows with $\Sigma_0^0(X) = \{\emptyset\}$ and $\Pi_0^0(X) = \{X\}$.

- $\Sigma_1^0(X)$ is the class of open subsets of X ,
- $\Pi_n^0(X)$ contains complements of sets from $\Sigma_n^0(X)$,
- $\Sigma_{n+1}^0(X)$ contains countable unions of sets from $\Pi_n^0(X)$.

The classes defined above are closed under inverse images of continuous functions. Given a classe \mathcal{C} , a set U is called \mathcal{C} -hard if each set in \mathcal{C} is an inverse image of U under some continuous function. If additionally $U \in \mathcal{C}$, U is said to be \mathcal{C} -complete. It is well known that every weakly recognizable tree language is a member of a Borel class of finite rank ([7,13]). The rank of a language is the rank of the minimal Borel class the language belongs to. It can be seen as a coarse measure of complexity of languages.

A much finer measure of the topological complexity is the *Wadge degree*. If $T, U \subseteq T_\Sigma$, we say that T is *continuously (or Wadge) reducible* to U , if there exists a continuous function f such that $T = f^{-1}(U)$. We write $T \leq_w U$ iff T is continuously reducible to U . Thus, given a certain Borel class \mathcal{C} , T is \mathcal{C} -hard if $U \leq_w T$ for every $U \in \mathcal{C}$. This particular ordering is called the *Wadge ordering*. If $T \leq_w U$ and $U \leq_w T$, then we write $T \equiv_w U$. If $T \leq_w U$ but not $U \leq_w T$, then we write $T <_w U$. The Wadge hierarchy is the partial order induced by $<_w$ on the equivalence classes given by \equiv_w .

Let T and U be two arbitrary sets of full binary trees. The *Wadge game* $\mathcal{W}(T, U)$ is a two-player game (player I and player II). During a play, each player builds a tree, say t_I and t_{II} . In each round both players add a finite number of children to the terminal nodes of their corresponding tree. Player I plays first and Player II is allowed to skip his turn but not forever. Player II wins the game iff $t_I \in T \Leftrightarrow t_{II} \in U$. Bill Wadge designed this game precisely in order to obtain a characterisation of continuous reducibility.

Lemma 1 ([23]). *Let $T, U \subseteq T_\Sigma$. Then $T \leq_w U$ iff Player II has a winning strategy in the game $\mathcal{W}(T, U)$.*

A language L is called *self dual* if it is equivalent to its complement, otherwise it is called *non self dual*. From Borel determinacy, if $T, U \subseteq T_\Sigma$ are Borel, then $\mathcal{W}(T, U)$ is determined. As a consequence, a variant of Martin-Monk's result shows that $<_w$ is well-founded. The *Wadge degree* for sets of finite Borel rank is inductively defined by:

- $d_w(\emptyset) = d_w(\emptyset^c) = 1$,
- $d_w(L) = \sup\{d_w(K) + 1 : K \text{ non self dual}, K <_w L\}$ for $L >_w \emptyset$.

2.3 Linear game automata

A *linear game automaton* (LGA) is a weak alternating automaton $A = \langle \Sigma, Q, q_I, \delta, \text{rank} \rangle$ satisfying two special restrictions:

- (game alternation) the transition relation is a total function $\delta : Q \times \Sigma \rightarrow Q \times Q$;
- (linearity) for every loop $q \xrightarrow{\sigma_0, d_0} q_1 \xrightarrow{\sigma_1, d_1} q_2 \cdots q_n \xrightarrow{\sigma_n, d_n} q$ it holds that $q_i = q$, for all $1 \leq i \leq n$.

In the remaining of the paper, we often write $q \xrightarrow{\sigma} q_0, q_1$ if $\delta(q, \sigma) = (q_0, q_1)$. And without loss of generality, we make the following assumptions:

- there is no trivial state, i.e., if $q \in Q$ is such that $A_q \equiv \top$ (resp. $A_q \equiv \perp$), then $q = \top$ (resp. $q = \perp$),
- there is no trivial transition, i.e., if $p \in Q_\forall$, and $p \xrightarrow{\sigma} q, \perp$, then $q = \perp$ (dually for $p \in Q_\exists$).

By convention, \top is a looping state of even rank, and \perp is a looping state of odd rank.

LGA are closed under complementation. The usual complementation procedure, that increases the ranks by one and swaps existential and universal states turns LGA into LGA. However, LGA are not closed under union nor intersection. Given $\sigma \in \Sigma$, the language $L_\sigma = \{t \in T_\Sigma : t(0) = t(1) = \sigma\}$ is LGA-recognizable, but $L_\sigma \cup L_{\sigma'}$ is not.

2.4 A normal form

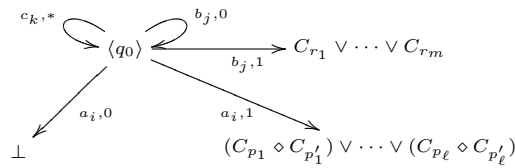
We now provide a useful normal form of LGA. First let us define three operations on tree languages (and tree automata). Let L, M be tree languages over Σ containing at least two letters, a and b . Define *alternative* (\vee), *disjunctive product* (\diamond), and *conjunctive product* (\square) as

$$\begin{aligned} L \vee M &= \{t : t(\varepsilon) = a, t.0 \in L \text{ or } t(\varepsilon) \neq a, t.0 \in M\}, \\ L \diamond M &= \{t : t.0 \in L \text{ or } t.1 \in M\}, \\ L \square M &= \{t : t.0 \in L \text{ and } t.0 \in M\}. \end{aligned}$$

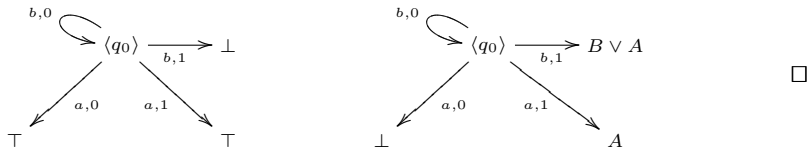
The family of languages recognized by LGAs is closed under these three operations. In particular, the operations have natural counterparts on automata. We write $A \vee B$ to denote the automaton recognizing $L(A) \vee L(B)$, and similarly for \diamond and \square . Multifold alternatives are performed from left to right, e.g., $A_1 \vee A_2 \vee A_3 \vee A_4 = (((A_1 \vee A_2) \vee A_3) \vee A_4)$. It is easy to see that these three operations define associative and commutative operations on Wadge equivalence classes.

Lemma 2. *Each LGA is Wadge equivalent to an LGA over the alphabet $\{a, b\}$.*

Proof. We proceed by induction on the number of states. Let C be an LGA. If C has only one state, the claim follows trivially. Suppose C has several states. We may assume w.l.o.g. that its initial state of C , q_0 , is existential. Suppose that the transitions of C starting in q_0 are $q_0 \xrightarrow{a_i} p_i, p'_i$, $q_0 \xrightarrow{b_j} q_0, r_i$ and $q_0 \xrightarrow{c_k} q_0, q_0$ with $\Sigma = \{a_1, \dots, a_\ell; b_1, \dots, b_m; c_1, \dots, c_n\}$. Then the automaton C is Wadge equivalent to the following automaton where C_q is the automaton obtained from C by changing the initial state to q .



By induction hypothesis, there exist automata A_i, A'_i, B_j over $\{a, b\}$, such that $A_i \equiv_w C_{p_i}$, $A'_i \equiv_w C_{p'_i}$, and $B_j \equiv_w C_{r_j}$. Let $A = (A_1 \diamond A'_1) \vee \dots \vee (A_\ell \diamond A'_\ell)$ and $B = B_1 \vee \dots \vee B_m$. Then, if $A \vee B \equiv_w \top$, the previous automaton is Wadge equivalent to the following automaton on the left and otherwise to the one on the right.

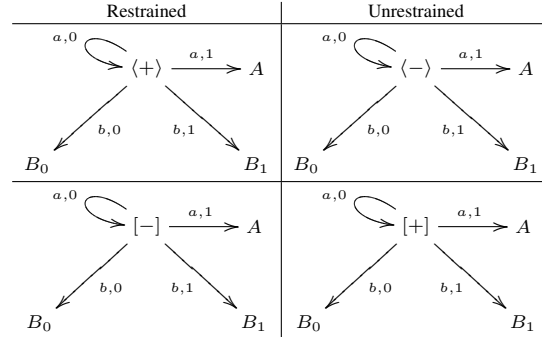


From now on we work with automata over $\{a, b\}$, unless explicitly stated otherwise.

A looping state q of an LGA A is

- *restrained* if it is an existential positive state or a universal negative state,
- *unrestrained* if it is an existential negative state or a universal positive state.

Examining the proof of Lemma 2, we see that in fact, each nontrivial looping state falls into exactly one of the categories shown below (+ means even rank, – means odd rank).



A node q of each of the above kinds may be seen as an action over triples of LGAs; we denote by $q(A, B_0, B_1)$ the automaton being the result of the action q on A, B_0, B_1 , e.g., $[+](A, B_0, B_1)$ or $\langle - \rangle(A, B_0, B_1)$. Often we use a shorthand $[\mu](A, B) = [\mu](A, B, \top)$, $\langle \mu \rangle(A, B) = \langle \mu \rangle(A, B, \perp)$ for $\mu = +$ or $\mu = -$.

3 Deciding the Borel hierarchy

3.1 Weak patterns menagerie

The basis for the procedure computing the Borel rank of a given LGA-recognizable language is a characterization in terms of difficult patterns. We define *weak* $(0, n)$ -*pattern*, and *weak* $(1, n + 1)$ -*pattern* by induction on n :

- a weak $(0, 1)$ -pattern is a negative loop reachable from a positive loop,
- a weak $(1, 2)$ -pattern is a positive loop reachable from a negative loop,
- a weak $(0, n + 1)$ -pattern is a weak $(1, n + 1)$ -pattern replicated by a universal positive node,
- a weak $(1, n + 2)$ -pattern is a weak $(0, n)$ -pattern replicated by an existential negative node.

We also define canonical automata, K_n^Σ and K_n^Π , corresponding to the weak patterns:

$$\begin{aligned}
 K_1^\Pi &= [+](\top, \perp, \perp), & K_{n+1}^\Pi &= [+](K_n^\Sigma, \perp, \perp), \\
 K_1^\Sigma &= \langle - \rangle(\perp, \top, \top), & K_{n+1}^\Sigma &= \langle - \rangle(K_n^\Pi, \top, \top).
 \end{aligned}$$

The tree languages recognized by the above canonical automata coincide with the sets used by Skurczyński to prove the existence of weakly recognizable languages of each finite Borel rank.

Proposition 1 ([22]). For every $n > 0$,
 $L(K_n^\Sigma)$ is Σ_n^0 -complete and $L(K_n^\Pi)$ is Π_n^0 -complete.

Skurczyński's result follows by straightforward induction from the following easy lemma. For $v \in \{0, 1\}^*$ and $U \subseteq T_\Sigma$, let $vU = \{t \in T_\Sigma : t.v \in U\}$.

Lemma 3. For each $n > 0$

1. if U_i is Σ_n^0 -hard for $i < \omega$, $\bigcap_{i \in \omega} 0^i 1U_i$ is Π_{n+1}^0 -hard;
2. if V_i is Π_n^0 -hard for $i < \omega$, $\bigcup_{i \in \omega} 0^i 1V_i$ is Σ_{n+1}^0 -hard.

3.2 Effective characterization

Since any Borel class is closed under finite unions and finite intersections, we have:

Proposition 2. Let K be a complete set for some class from $\bigcup_{1 \leq i < \omega} \{\Sigma_i^0, \Pi_i^0\}$.
For every k , if $U_i \leq_W K$ for $0 \leq i \leq k$, then

$$(1) \bigcup_{i=0}^k 0^i 1U_i \leq_W K, \quad (2) \bigcap_{i=0}^k 0^i 1U_i \leq_W K,$$

and if $V_i <_W K$ for $0 \leq i \leq k$, then

$$(3) \bigcup_{i=0}^k 0^i 1V_i <_W K, \quad (4) \bigcap_{i=0}^k 0^i 1V_i <_W K.$$

Analogously, since Σ_n^0 is closed under countable unions, and Π_n^0 is closed under countable intersections, we obtain the following result.

Proposition 3.

1. Let K be a Σ_n^0 -complete set. If for every $i \in \omega$ it holds that $U_i \leq_W K$, then $\bigcup_{i \in \omega} 0^i 1U_i \leq_W K$.
2. Let K be a Π_n^0 -complete set. If for every $i \in \omega$ it holds that $U_i \leq_W K$, then $\bigcap_{i \in \omega} 0^i 1U_i \leq_W K$.

We now apply these properties to characterize the topological power of looping nodes in an LGA.

Lemma 4. Let A, B_0, B_1, C be LGA such that $C = q(A, B_0, B_1)$, and q is a restrained looping node. For $n \geq 2$

1. if $L(A), L(B_i) <_w L(K_n^\Sigma)$, then $L(C) <_w L(K_n^\Sigma)$;
2. if $L(A), L(B_i) <_w L(K_n^\Pi)$, then $L(C) <_w L(K_n^\Sigma)$.

Proof. It is enough to prove the first claim, the second follows by duality. Suppose that $q = \langle + \rangle$. Let us describe a winning strategy for Player II in $\mathcal{W}(L(A), L(K_n^\Sigma))$. If Player I plays a on the leftmost branch, Player II plays accepting in the subtrees rooted in nodes $0^i 1$. If Player I finally plays a b in the k th round, Player II switches to playing rejecting in every subtree rooted in $0^i 1$ for $i < k$, and in the subtree rooted in 0^k applies the winning strategy given by Proposition 2 (1). Hence, $L(C) \leq_w L(K_n^\Sigma)$.

To obtain strictness of the inequality, we describe a winning strategy for Player I in $\mathcal{W}(L(K_n^\Sigma), L(A))$. As long as Player II skips or plays a on the leftmost branch, Player I plays rejecting in the subtrees rooted in $0^i 1$. If in the k th round Player II finally plays

b on the leftmost branch, Player I continues playing rejecting in every subtree rooted in $0^i 1$ for $i \leq n$, and in the subtree rooted in 0^{n+1} applies the winning strategy given by Proposition 2 (3).

For $q = [-]$ the proof is analogous, only uses Proposition 2 (2) and (4). \square

Lemma 5. *Let A, B_0, B_1, C be LGA such that $C = q(A, B_0, B_1)$, and q is an unrestrained looping node. Let $n \geq 2$. If $q = \langle - \rangle$, then*

1. *if $L(A) \leq_w L(K_{n-1}^\Sigma)$, and $L(B_i) <_w L(K_n^\Sigma)$, then $L(C) <_w L(K_n^\Sigma)$;*
2. *if $L(A) \geq_w L(K_{n-1}^\Pi)$, then $L(C) \geq_w L(K_n^\Sigma)$;*

and if $q = [+]$, then

3. *if $L(A) \leq_w L(K_{n-1}^\Pi)$, and $L(B_i) < L(K_n^\Pi)$, then $L(C) <_w L(K_n^\Pi)$;*
4. *if $L(A) \geq_w L(K_{n-1}^\Sigma)$, then $L(C) \geq_w L(K_n^\Pi)$.*

Proof. Use an argument similar to the proof of Lemma 4 to infer (1) and (3) from Proposition 3, and (2) and (4) from Lemma 3. \square

The main theorem of this part follows from Lemma 4 and Lemma 5 by induction on the structure of the automaton. And as a corollary, we obtain the first decidability result.

Theorem 1. *For every n and every LGA A*

1. *$L(A)$ is Σ_n^0 -hard iff A contains a weak $(1, n+1)$ -pattern;*
2. *$L(A)$ is Π_n^0 -hard iff A contains a weak $(0, n)$ -pattern.*

Corollary 1. *The problem of calculating the exact position in the Borel hierarchy of a language recognized by a linear game tree automaton is decidable (in polynomial time if the productive states are given).*

4 The weak index hierarchy

4.1 Introducing the hierarchy

The (Mostowski–Rabin) index of an automaton A is given by $(i, j) \in \omega \times \omega$, where i is the minimal and j is the maximal value of the priority function rank. Scaling down the priorities if necessary, we can assume that $i \in \{0, 1\}$ and that for every $n \in \{i, i+1, \dots, j\}$, there is a state q such that $\text{rank}(q) = n$. Thus, the indices are elements of $(\{0, 1\} \times \omega) \setminus (1, 0)$. Given an index $(0, j)$ (resp. $(1, j)$), its dual index is $(1, j+1)$ (resp. $(0, j-1)$).

Consider the partial order on indices of automata given by

$$(i, j) \sqsubseteq (i', j') \quad \text{iff} \quad j - i < j' - i'.$$

Note that this implies that dual indices are incomparable. The hierarchy induced by the partial order \sqsubseteq is called the *hierarchy of Mostowski–Rabin indices* (or simply the index hierarchy) of the considered class of automata.

For a given class, the hierarchy is said to be strict if there is an automaton at each level that cannot be simulated by any automaton from the same class of lower level. By a result of Bradfield [3,4], we know that the index hierarchy of alternating tree automata, and therefore the fixpoint hierarchy of the modal μ -calculus, is strict. Arnold's proof of the same result [1] can be adapted to show that the index hierarchy of weak alternating tree automata is also strict. In the latter case we speak of the *weak index hierarchy*.

4.2 The conjecture

In [16] it was conjectured that for weakly recognizable tree languages the weak index hierarchy and the Borel hierarchy coincide, i.e., that a weakly recognizable tree language is in Σ_n^0 (resp. Π_n^0) iff it can be recognized by a weak alternating automaton of index $(1, n+1)$ (resp. $(0, n)$). It has long been known that one implication holds.

Proposition 4 ([13]). *For every weak alternating automaton with index $(0, n)$ (resp. $(1, n+1)$), it holds that $L(A) \in \Pi_n^0$ (resp. $L(A) \in \Sigma_n^0$).*

It was also proved recently that the conjecture holds when restricted to languages which are in addition deterministically recognizable [16]. We refine this result by showing that the conjecture also holds for languages recognizable by LGA.

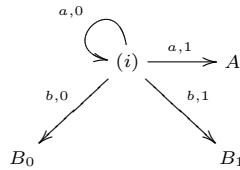
4.3 Weak index of LGA-recognizable sets

Theorem 2. *For languages recognizable by LGA, the Borel hierarchy and the weak index hierarchy coincide.*

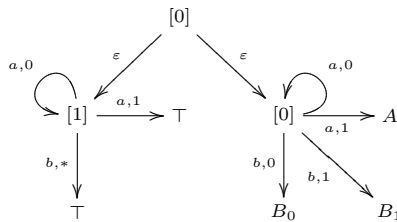
Proof. By duality it is enough to consider Π_n^0 classes. By Proposition 4 it suffices to show that for each LGA C with $L(A) \in \Pi_n^0$ there exists an equivalent weak alternating automaton of index $(0, n)$. We proceed by induction on the structure of the automaton.

The case $n = 0$ is trivial. Suppose that $n = 1$. By Theorem 1, C does not contain an accepting loop reachable from a rejecting loop. It is enough to set the rank of all states reachable from odd looping states to 1 and the rank of the remaining states to 0 to obtain an equivalent automaton of index $(0, 1)$.

Suppose that $n \geq 2$. If the initial state of C is not looping, the claim follows easily from the induction hypothesis. Suppose that q_0 is a looping node, and C is of the form

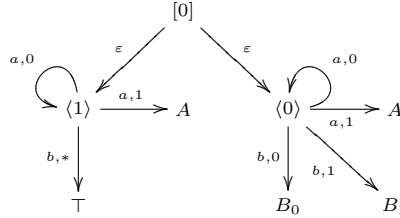


We can treat C as a weak alternating automaton and transform it into an equivalent one of index $(0, n)$. Clearly, it must hold that $L(A), L(B) \in \Pi_n^0$ and by induction hypothesis we may assume that A, B_0, B_1 have index $(0, n)$. If $i = 0$, the claim follows trivially. If $(i) = [1]$, the equivalent weak alternating automaton of index $(0, n)$ is shown below.



To prove the equivalence, observe that the left-hand component checks that finally b occurs on the leftmost branch, and the right-hand component checks the condition A until the first b occurs, and after that checks the conditions B_0 and B_1 .

Finally, suppose that $(i) = \langle 1 \rangle$. By Theorem 1, C contains no weak $(1, n + 1)$ -pattern, which implies that A contains no weak $(0, n - 1)$ -pattern. By induction hypothesis we may assume that A has index $(1, n)$. Recall that B_0 and B_1 have index $(0, n)$. The corresponding equivalent weak alternating automaton is shown below.



The left-hand component takes care of the situation, when b never occurs on the leftmost path. If b does occur, it is trivially accepting, but the right-hand component provides the appropriate semantics. \square

Combining Theorem 1 and Theorem 2 we obtain the second decidability result.

Corollary 2. *The problem of calculating the exact position in the weak index hierarchy of a language recognized by a LGA is decidable (in polynomial time if the productive states are given).*

5 The Wadge hierarchy

5.1 The difference hierarchy

For a Borel class Σ_n^0 , the finite Hausdorff-Kuratowski, or difference, hierarchy is defined as $\text{Diff}_1(\Sigma_n) = \Sigma_n$ and $\text{Diff}_k(\Sigma_n) = \{U \setminus V : U \in \Sigma_n, V \in \text{Diff}_{k-1}(\Sigma_n)\}$. Let $\overline{\text{Diff}}_k(\Sigma_n)$ denote the dual class. Recall that this is not the same as $\text{Diff}_k(\Pi_n)$. Indeed, $\text{Diff}_{2k+1}(\Pi_n) = \overline{\text{Diff}}_{2k+1}(\Sigma_n)$ and $\text{Diff}_{2k}(\Pi_n) = \text{Diff}_{2k}(\Sigma_n)$. We have

$$\begin{aligned} \text{Diff}_{2k}(\Sigma_n) &= \{U_1 \cap V_1^c \cup \dots \cup U_k \cap V_k^c\}, \\ \text{Diff}_{2k+1}(\Sigma_n) &= \{U_1 \cap V_1^c \cup \dots \cup U_k \cap V_k^c \cup U\}, \\ \overline{\text{Diff}}_{2k}(\Sigma_n) &= \{U_1 \cap V_1^c \cup \dots \cup U_{k-1} \cap V_{k-1}^c \cup U \cup V^c\}, \\ \overline{\text{Diff}}_{2k+1}(\Sigma_n) &= \{U_1 \cap V_1^c \cup \dots \cup U_k \cap V_k^c \cup V^c\}, \end{aligned}$$

where the sets U, V, U_i, V_i range over Σ_n . From this characterization one easily obtains the following table of the operation \diamond . For $n > 0$ let $S_n(k)$ be a $\text{Diff}_k(\Sigma_n)$ -complete set, and let $P_n(k)$ be a $\overline{\text{Diff}}_k(\Sigma_n)$ -complete set.

Lemma 6. For each $n > 0$, $i > 0$, $j \geq 0$

- $S_n(2i) \diamond S_n(2j) \equiv S_n(2i + 2j)$
 - $S_n(2i + 1) \diamond S_n(2j) \equiv S_n(2i + 2j + 1)$,
 - $S_n(2i) \diamond P_n(2j) \equiv P_n(2i + 2j)$
 - $S_n(2i + 1) \diamond P_n(2j) \equiv P_n(2i + 2j)$,
 - $P_n(2i) \diamond S_n(2j) \equiv S_n(2i + 2j)$
 - $P_n(2i + 1) \diamond S_n(2j) \equiv P_n(2i + 2j + 1)$,
 - $P_n(2i) \diamond P_n(2j) \equiv P_n(2i + 2j - 2)$
 - $P_n(2i + 1) \diamond P_n(2j) \equiv P_n(2i + 2j)$,
- $S_n(2i + 1) \diamond S_n(2j + 1) \equiv S_n(2i + 2j + 1)$,
 - $S_n(2i + 1) \diamond P_n(2j + 1) \equiv P_n(2i + 2j + 2)$,
 - $P_n(2i + 1) \diamond S_n(2j + 1) \equiv P_n(2i + 2j + 2)$,
 - $P_n(2i + 1) \diamond P_n(2j + 1) \equiv P_n(2i + 2j + 1)$.

The equivalences above, together with closure by \diamond , immediately provide complete LGA-recognizable languages for $\text{Diff}_k(\Sigma_n)$ for each k, n . Building upon this we produce the whole Wadge hierarchy of LGA-recognizable languages.

5.2 Bestiarum vocabulum

For an ordinal α let $\exp(\alpha) = \omega_1^\alpha$. Hence,

$$\exp^{k+1}(\alpha) = \exp(\exp^k(\alpha)) = \underbrace{\omega_1^{\omega_1^{\dots^{\omega_1^\alpha}}}}_{k+1 \text{ times } \omega_1}.$$

Before describing the hierarchy, recall the Wadge degrees of $\text{Diff}_k(\Sigma_n)$ -complete sets.

Proposition 5 ([5]). For each $k > 0$, $d_w(S_n(k)) = d_w(P_n(k)) = \exp^n(k)$.

Theorem 3. The family of LGA-recognizable languages contains L with $d_w(L) = \beta$ for every $\beta = \sum_{i=n}^0 \beta_i$, where each β_i is of the form

$$\beta_i = \exp^i(\omega)\eta + \sum_{p=j}^1 \exp^i(p)k_p$$

with $\eta < \omega^\omega$, $k_{2q} \in \{0, 1\}$, and $j, k_{2q+1} < \omega$.

Proof. By induction on such ordinals, we provide an automaton A_β , such that $L(A_\beta)$ is non self dual, and $d_w(L(A_\beta)) = \beta$. To make the notation more readable, we use bracketed ordinal $[\beta]$ to denote the automaton A_β . Since LGA are closed under complementation, when we construct an automaton recognizing a non self dual set of degree β , we also immediately get the automaton $[\beta]^c$. We write $[\beta]^\pm$ for $[\beta] \vee [\beta]^c$.

Let us start with the basic building bricks of our construction: the automata $[1]$, $[\omega^m]$, $[\exp^i(1)]$, and $[\exp^i(\omega)\omega^p]$. Together with these automata we show how to make a step with those ordinals, i.e., how to define the automaton for $[\alpha + \gamma]$, once we already have the automaton $[\alpha]$ and γ is one of the above. Let

$$[1] = \perp, \quad [\alpha + 1] = \langle + \rangle(\perp, [\alpha]^\pm).$$

Note that $[2] = K_1^\Sigma$, and $[2]^G = K_1^\Pi$. For $m > 1$ let

$$\begin{aligned} [\omega] &= \langle + \rangle([3], \perp), & [\alpha + \omega] &= \langle + \rangle([3], [\alpha]^\pm), \\ [\omega^m] &= \langle + \rangle([\omega^{m-1} + 1], \perp), & [\alpha + \omega^m] &= \langle + \rangle([\omega^{m-1} + 1], [\alpha]^\pm). \end{aligned}$$

For $i > 1$ let

$$\begin{aligned} [\exp(1)] &= \langle - \rangle([2]^G, \perp), & [\alpha + \exp(1)] &= \langle - \rangle([2]^G, [\alpha]^\pm), \\ [\exp^i(1)] &= \langle - \rangle([\exp^{i-1}(1)]^G, \perp), & [\alpha + \exp^i(1)] &= \langle - \rangle([\exp^{i-1}(1)]^G, [\alpha]^\pm). \end{aligned}$$

Note that $[\exp^i(1)] = K_{i+1}^\Sigma$, $[\exp^i(1)]^G \equiv K_{i+1}^\Pi$. For $p > 0$ let

$$\begin{aligned} [\exp^i(\omega)] &= \langle + \rangle([\exp^i(2)], \perp), \\ [\alpha + \exp^i(\omega)] &= \langle + \rangle([\exp^i(2)], [\alpha]^\pm), \\ [\exp^i(\omega)\omega^p] &= \langle + \rangle([\exp^i(\omega)\omega^{p-1} + 1], \perp), \\ [\alpha + \exp^i(\omega)\omega^p] &= \langle + \rangle([\exp^i(\omega)\omega^{p-1} + 1], [\alpha]^\pm). \end{aligned}$$

Using the basic building blocks and basic steps defined above we can inductively define automata $[\sum_{i=n}^1 \gamma_i]$, such that each δ_i is of the form $\exp^i(\omega)\eta + \exp^i(1)p$ with $\eta < \omega^\omega$ and $p < \omega$.

To define automata for all β described in the statement of the theorem, we need one more kind of bricks and two more kinds of steps. For $\eta < \omega^\omega$, $1 \leq i < \omega$, we have:

$$\begin{aligned} [\exp^i(2)] &= [\exp^i(1)] \diamond [\exp^i(1)]^G \\ [\alpha + \exp^i(\omega)\eta + \sum_{p=m}^1 \exp^i(p+2)k_p] &= [\alpha + \exp^i(\omega)\eta + \sum_{p=m}^1 \exp^i(p)k_p] \diamond [\exp^i(2)] \\ [\alpha + \exp^i(\omega)\eta + \sum_{p=m}^1 \exp^i(p+2)k_p + \exp^i(2)] &= [\alpha + \exp^i(\omega)\eta + \sum_{p=m}^{\ell} \exp^i(p)k_p + 1] \diamond [\exp^i(2)]. \end{aligned}$$

Using Lemma 6 and standard Wadge game arguments one can prove that for every ordinal α from the statement of the theorem, $[\alpha]$ has Wadge degree α . \square

As a corollary we obtain a lower bound on the height of the hierarchy.

Corollary 3. *The LGA hierarchy has height at least $(\omega^\omega)^\omega = \omega^{\omega^2}$.*

In the remaining of the paper we prove that the height of the LGA hierarchy is exactly $(\omega^\omega)^\omega$ and that we can compute the Wadge degree of a language LGA-recognizable.

5.3 Two simple operations on sets of trees

Let us define two more operations on sets of trees. Let $L, M \subseteq T_\Sigma$, $a, b \in \Sigma$. We define the set $L \rightarrow M$ as the set of trees $t \in T_\Sigma$, satisfying any of the following conditions:

- $t.1 \in L$ and $a = t(0^n)$ for all n ,
- 00^n is the first node on the branch 00^* such that $a \neq t(00^n)$ and $t.00^{n-1} \in M$.

A player in charge of $L \rightarrow M$ is like a player in charge of L endowed with an extra move, which can be used only once, that erases everything played before. Then he can restart the play being in charge of M .

The second operation is a generalization of \vee . Let $L_n \subseteq T_\Sigma$ for $n < \omega$. Define $\sup_{n < \omega}^- L_n$ as the set of trees $t \in T_\Sigma$ satisfying the following conditions for some k :

- 0^k is the first node on 1^* labeled with b ,
- $t.0^k 1 \in L_k$.

Intuitively, a player in charge of $\sup_{n < \omega}^- L_n$ is given the choice between the L_n 's. The decision is determined by the number of a 's played on the leftmost branch of the tree before the first b . If the player keeps playing a 's forever on the leftmost branch, the tree will be rejected.

Define also $\sup_n^+ L_n$ as $\sup_n^- L_n \cup \{t : \forall_n t(1^n) = a\}$. The difference from the previous operation is that now, when the player plays a 's forever on the leftmost branch, the obtained tree is accepted. Note that the operations are dual:

$$\left(\sup_n^+ L_n \right)^c = \sup_n^- \left(L_n^c \right)$$

5.4 Computing Wadge degrees

Let Ω denote the set of Wadge equivalence classes of languages recognized by the automata $[\beta]$, $[\beta]^c$, $[\beta]^\pm$ defined in the proof of Theorem 3. Slightly abusing the notation we write $[\beta]^-$ for the Wadge equivalence class of $L([\beta])$, $[\beta]^+$ for the class of $L([\beta]^c)$, and $[\beta]^\pm$ for the class of $L([\beta]^\pm)$.

The technical difficulty of the decidability result lies in the following effective closure property (its proof can be found in the appendix).

Theorem 4. *For each $U, V \in \Omega$ it holds that $U \diamond V$, $U \vee V$, and $\sup_k^+ U^{(k)} \diamond V$ belong to Ω and can be effectively computed. The same holds for $U \rightarrow V$, if $U = [\exp^i(1)]^\mu$ for some $i < \omega$ and $\mu \in \{+, -\}$.*

Theorem 5. *For each LGA we can calculate effectively the signed degree of the recognized language.*

Proof. We proceed by induction on the number of states. Let C be an LGA. If C has only one state, it is either totally accepting or totally rejecting. In the first case the signed degree is $[1]^+$, in the second case it is $[1]^-$. Suppose that C has more states. By duality we may assume that the initial state q_0 is existential: if it is universal, compute the signed degree for the complement of C , and return the degree negated. Suppose that q_0 is not looping. By linearity, C can be represented as in Figure 1(a) for some automata A_0, A_1, B_0, B_1 , each having less states than C . Clearly $L(C) \equiv L(A_0) \diamond L(A_1) \vee L(B_0) \diamond L(B_1)$. Hence, we can use the induction hypothesis to get the degrees of $L(C_{q_i})$, and then Theorem 4 to compute $d(C) = d(C_{q_1}) \diamond d(C_{q_2}) \vee d(C_{q_3}) \diamond d(C_{q_4})$.

If q_0 is looping, we can assume w.l.o.g. that C is given in Figure 1(b) with $i = 0, 1$. If $i = 1$. There exists $n \in \omega$ such that $L(A)$ is either Σ_n^0 -complete, or in $\Delta_{n+1}^0 \setminus \Sigma_n^0$. If $L(A)$ is Σ_n^0 -complete, by Lemma 5, the language recognized by C' , defined in Figure 1(c) is also Σ_n^0 -complete. Since $d(A) = d(C') = [\exp^n(1)]^-$ and $([\exp^n(1)]^-)^{\langle k \rangle} = [\exp^n(1)]^-$ for each $k > 0$, we have $d(C) = [\exp^n(1)]^- \rightarrow d(B_1) \diamond (B_2) \diamond [\exp^n(1)]^-$. On the other hand, if $L(A) \in \Delta_{n+1}^0 \setminus \Sigma_n^0$, by Theorem 1 and Theorem 2, the language recognized by C' is Σ_{n+1}^0 -complete, and it is easy to see that $d(C) = [\exp^{n+1}(1)]^- \rightarrow d(B_1) \diamond d(B_2)$. We conclude by the inductive hypothesis and Theorem 4.

If $i = 0$, it is straightforward to check that $d(C) = \sup_k^+ d(A)^{\langle k \rangle} \diamond d(B)$, and again the claim follows from Theorem 4 and the induction hypothesis. \square

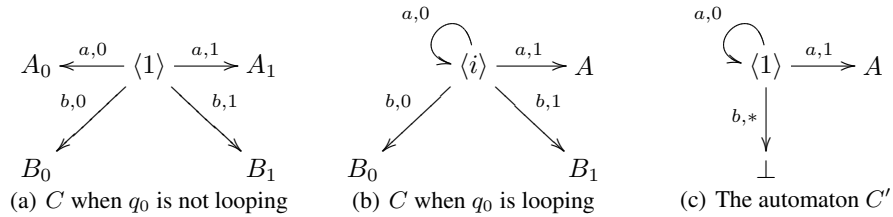


Fig. 1. The automata C and C'

6 Conclusion

Alternating tree automata are notorious for the lack of decision procedures for classical hierarchies like Mostowski-Rabin hierarchy, Borel hierarchy, or Wadge hierarchy. The reason for this is that when we move from infinite words to infinite trees, deterministic and non-deterministic modes of computation highly diverge.

We have proposed a novel class of automata capturing an interesting aspect of alternation, and for this class we have proved that all corresponding hierarchies mentioned above are decidable. Moreover we have shown that the weak index and the Borel rank coincide over LGA-recognizable automata.

We have seen that, despite their apparent simplicity, LGA yield a class of languages surprisingly complex from the topological point of view: the height of their Wadge hierarchy is $(\omega^\omega)^\omega$. Admittedly, this is much less than the hierarchy for weak alternating automata, which is known to be at least ϵ_0 high [7]. But this was expected, as LGA form a very restricted subclass of weak alternating automata. What is surprising however, is that the height of the Wadge hierarchy for LGA is much larger than that for deterministic automata, which was shown in [15] to be $(\omega^\omega)^3 + 3$, and the same as for deterministic *push-down* automata on infinite words [6].

Acknowledgment. We thank Igor Walukiewicz and Claire David for helpful comments and suggestions, David Janin for initiating this research by showing the interest in the topological complexity of weak alternating automata, Sławek Lasota for bringing linear automata into our attention.

References

1. A. Arnold. The μ -Calculus Alternation-Depth Hierarchy is Strict on Binary Trees. *ITA* 33(4/5): 329–340 (1999).
2. A. Arnold, D. Niwiński. Continuous Separation of Game Languages. *Fundamenta Informaticae*, 81(1–3): 19–28 (2008).
3. J. Bradfield. The Modal μ -Calculus Alternation Hierarchy is Strict. *Theor. Comput. Sci.* 195(2): 133–153 (1998).
4. J. Bradfield. Simplifying the Modal μ -Calculus Alternation Hierarchy. *STACS 1998*: 39–49 (1998).
5. J. Duparc. Wadge Hierarchy and Veblen Hierarchy Part 1: Borel Sets of Finite Rank. *J. Symb. Log.* 66(1): 56–86 (2001).
6. J. Duparc. A Hierarchy of Deterministic Context-Free ω -Languages. *Theoret. Comput. Sci.* 290:1253–1300 (2003).
7. J. Duparc, F. Murlak. On the Topological Complexity of Weakly Recognizable Tree Languages. *FCT 2007*, LNCS 4639: 261–273 (2007).
8. O. Finkel. Borel Ranks and Wadge Degrees of ω -Context Free Languages. *Mathematical Structures in Computer Science* 16: 813–840 (2006).
9. S. Hummel, H. Michalewski, D. Niwiński. On the Borel Inseparability of Game Tree Languages. *Proc. of STACS '09*: 565–576 (2009).
10. O. Kupferman, S. Safra, M. Vardi. Relating Word and Tree Automata. *LICS 1996*: 322–332 (1996).
11. O. Kupferman, M. Vardi, P. Wolper. An Automata-Theoretic Approach to Branching-Time Model Checking. *Journal of the ACM* 47(2): 142–155 (1994).
12. L. H. Landweber. Decision Problems for ω -Automata. *Math. Systems Theory* 3: 376–384 (1969).
13. A. W. Mostowski. Hierarchies of Weak Automata and Weak Monadic Formulas. *Theoret. Comput. Sci.* 83: 323–335 (1991)
14. F. Murlak. On Deciding Topological Classes of Deterministic Tree Languages. In *Proc. CSL '05*, LNCS 3634: 573–584 (2005)
15. F. Murlak. The Wadge Hierarchy of Deterministic Tree Languages. *Logical Methods in Comput. Sci.*, 4(4), Paper 15.
16. F. Murlak. Weak Index vs Borel Rank. In *Proc. STACS '08*: 573–584 (2008).
17. D. Niwiński. On Fixed Point Clones. In *Proc. ICALP '86*, LNCS 226: 464–473 (1986).
18. D. Niwiński, I. Walukiewicz. Relating Hierarchies of Word and Tree Automata. In *Proc. STACS '98*, LNCS 1373: 320–331 (1998).
19. D. Niwiński, I. Walukiewicz. A Gap Property of Deterministic Tree Languages. *Theor. Comput. Sci.* 303: 215–231 (2003).
20. D. Niwiński, I. Walukiewicz. Deciding Nondeterministic Hierarchy of Deterministic Tree Automata. *Electr. Notes Theor. Comput. Sci.* 123: 195–208 (2005).
21. V. Selivanov. Wadge Degrees of ω -Languages of Deterministic Turing Machines. *Theoret. Informatics Appl.*, 37: 67–83 (2003).
22. J. Skurczyński. The Borel Hierarchy is Infinite in the Class of Regular Sets of Trees. *Theoret. Comput. Sci.* 112: 413–418 (1993).
23. W. W. Wadge. *Reducibility and Determinateness on the Baire Space*. Ph.D. Thesis, Berkeley (1984).
24. K. Wagner. Eine topologische Charakterisierung einiger Klassen regulärer Folgenmengen. *J. Inf. Process. Cybern.* EIK 13: 473–487 (1977).
25. K. Wagner. On ω -Regular Sets. *Inform. and Control* 43: 123–177 (1979).

A Example

In figure 2, an example of a linear game automaton of index $(1, 6)$ is given. Figure 3 represents the computation of its Wadge degree, which turn out to be $\exp^2(\omega) + \exp^2(3)$. In this last figure each state is labeled with the equivalence class of the subautomaton with this very state as its initial one. In both figures, \dashrightarrow (resp. \rightarrow) stands for left (resp. right) transition.

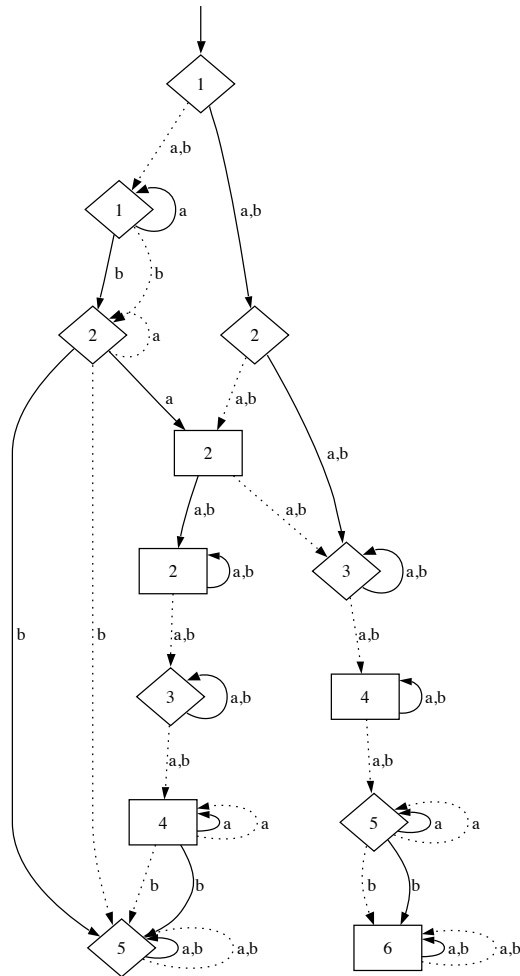


Fig. 2. A linear game automaton ...

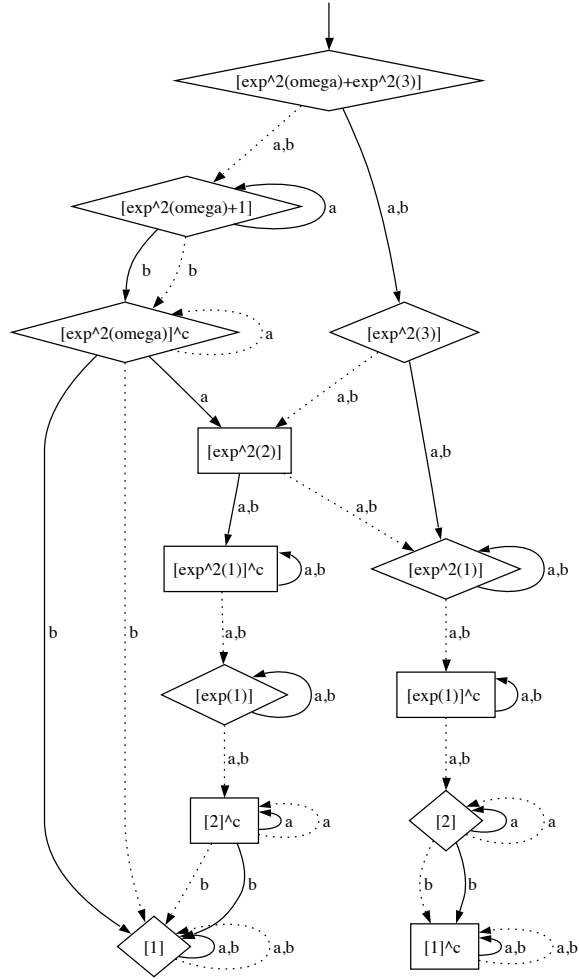


Fig. 3. ... and the computation of its Wadge degree.

B Proofs

B.1 Basic properties of operations

We say that a non self dual set L is *initializable* if $L \geq_w L \rightarrow L$. The following lemmas summarize simple yet useful properties of the operations on languages. They can be proved with standard Wadge game arguments.

Lemma 7. For initializable A, B and arbitrary A', B' and $A_n, B_n, n < \omega$

$$\begin{aligned}
(A \rightarrow A') \diamond (B \rightarrow B') &\equiv_w A \diamond B \rightarrow ((A \rightarrow A') \diamond B' \vee A' \diamond (B \rightarrow B')), \\
(A \rightarrow A') \diamond B &\equiv_w A \diamond B \rightarrow A' \diamond B, \\
(A \rightarrow A') \diamond \sup_n^+ B_n &\equiv_w \sup_n^+ (A \rightarrow A') \diamond B_n, \\
(A \rightarrow A') \diamond \sup_n^- B_n &\equiv_w A \rightarrow \left((A' \diamond \sup_n^- B_n) \vee (\sup_n^- (A \rightarrow A') \diamond B_n) \vee (\sup_n^+ (A \rightarrow A') \diamond B_n) \right) \quad \text{for } A > \perp, \\
(\perp \rightarrow A') \diamond \sup_n^- B_n &\equiv_w \perp \rightarrow \left((A' \diamond \sup_n^- B_n) \vee (\sup_n^- (\perp \rightarrow A') \diamond B_n) \right), \\
(\top \rightarrow A') \diamond \sup_n^- B_n &\equiv_w \top \rightarrow \left((A' \diamond \sup_n^- B_n) \vee (\sup_n^+ (\top \rightarrow A') \diamond B_n) \right), \\
\sup_m^+ A_m \diamond \sup_n^+ B_n &\equiv_w \top \rightarrow \left(\sup_m^+ (A_m \diamond \sup_n^+ B_n) \vee (\sup_n^+ (\sup_m^+ A_m) \diamond B_n) \right), \\
\sup_m^+ A_m \diamond \sup_n^- B_n &\equiv_w \top \rightarrow \left(\sup_m^+ (A_m \diamond \sup_n^- B_n) \vee (\sup_n^+ (\sup_m^+ A_m) \diamond B_n) \right), \\
\sup_m^- A_m \diamond \sup_n^- B_n &\equiv_w \perp \rightarrow \left(\sup_m^- (A_m \diamond \sup_n^- B_n) \vee (\sup_n^- (\sup_m^- A_m) \diamond B_n) \right).
\end{aligned}$$

Lemma 8. For arbitrary sets A, B, C, D it holds that

- $(A \vee B) \diamond C \equiv_w A \diamond C \vee B \diamond C$,
- if $A \leq_w C$ and $B \leq_w D$, then $A \diamond B \leq_w B \diamond D$.

Let us now recall the elegant relation between operations on sets and Wadge degrees. Let $M + L = L \rightarrow M \vee M^{\mathcal{C}}$.

Proposition 6 ([5]). For arbitrary Borel sets L, M, L_n with $n < \omega$ it holds that

$$\begin{aligned}
d_w(M + L) &= d_w(M) + d_w(L), \\
d_w(\sup_n^- L_n) &= d_w(\sup_n^+ L_n) = \sup_n (d_w(L_n) + 1).
\end{aligned}$$

B.2 Closure by \vee and \rightarrow

For $n > 0$, let $\hat{\Omega}_n$ denote the set of signed ordinals of the form $[\Sigma_{i=1}^N \exp^n(i)k_i]^\mu$ for $\mu \in \{+, -, \pm\}$ and some natural N, k_i , with $k_i \in \{0, 1\}$ for even i . For uniformity, $\hat{\Omega}_0 = \{0^+, 0^-, 0^\pm\}$. Let $\Omega_n = \{[\exp^n(\omega)\alpha + \beta]^\mu : \mu \in \{+, -, \pm\}, \alpha < \omega^\omega, \beta^\mu \in \hat{\Omega}_n\}$ for $n > 0$ and $\Omega_0 = \{[\alpha]^\mu : \mu \in \{+, -, \pm\}, \alpha < \omega^\omega\}$. Recall that by Ω we denote the set of signed ordinals of the form $[\alpha_k + \dots + \alpha_1 + \alpha_0]^\mu$, with $\alpha_i^\pm \in \Omega_i$.

The signing convention we adopted, in short, constitutes in assigning $+$ to the equivalence classes whose canonical represents were obtained as $\langle + \rangle(A, B)$, which corresponds to \sup^+ , and to those obtained as $\langle + \rangle(A, B)$, which results in $\text{Diff}_k(\Sigma_n)$ -complete sets being negative, and $\text{Diff}_k(\Sigma_n)$ -complete sets being positive.

The closure of Ω by \vee is very simple. If $[\alpha]^\mu$ and $[\beta]^\nu$ are comparable, $[\alpha]^\mu \vee [\beta]^\nu$ is simply equal to the larger of the two. If $[\alpha]^\mu$ and $[\beta]^\nu$ are incomparable, then necessarily $[\beta]^\nu = [\alpha]^\mu$ and the result is $[\alpha]^\pm$.

Let us now concentrate on \rightarrow . First we state yet another simple observation. By $\bar{\mu}$ we denote the dual sign:

$$\bar{\mu} = \begin{cases} + & \text{if } \mu = -, \\ - & \text{if } \mu = +, \\ \pm & \text{if } \mu = \pm. \end{cases}$$

Lemma 9. For arbitrary tree sets A_n ,

$$[1]^\mu \rightarrow \sup_n^\nu A_n \equiv_w \begin{cases} \sup_n^\nu A_n & \text{if } \mu = \nu \in \{-, +\} \\ [1]^\mu \rightarrow (\sup_n^+ A_n) \vee (\sup_n^- A_n) & \text{if } \mu = \bar{\nu} \in \{-, +\}. \end{cases}$$

Observe that by Proposition 6 $[\exp^i(1)]^+ \rightarrow [\beta]^\pm \equiv_w [\beta + \exp^i(1)]^+$. Thus, it follows that the result is in Ω . Consider now $[\exp^i(1)]^+ \rightarrow [\beta]^\nu$ for $\nu \in \{-, +\}$. If $[\beta]^\nu = B \rightarrow [\beta']^\pm$, with B initializable, it follows that $[\exp^i(1)]^\mu \rightarrow [\beta]^\nu \equiv_w ([\exp^i(1)]^+ \rightarrow B) \rightarrow [\beta']^\pm$. It is easy to see that

$$[\exp^i(1)]^+ \rightarrow B \equiv_w \begin{cases} [\exp^i(1)]^+ & \text{for } B \leq_w [\exp^i(1)]^+ \\ B & \text{for } B \geq_w [\exp^i(1)]^+, \\ [\exp^i(1)2]^+ & \text{for } B \equiv_w [\exp^i(1)]^-, \end{cases}$$

and we can conclude from the previous case. The remaining case is that of $[\beta]^\nu = \sup^\nu[\beta' + \beta_n]^+$, with $\beta_n = \exp^i(\omega)\omega^n$ or $\beta_n = \exp^i(n)$. If $i > 0$, we have $[\exp^i(1)]^\mu \rightarrow [\beta]^\nu \equiv_w ([\exp^i(1)]^\mu \rightarrow [1]^\nu) \rightarrow [\beta]^\nu \equiv_w [\exp^i(1)]^\mu \rightarrow ([1]^\nu \rightarrow [\beta]^\nu)$. By Lemma 9, this is equal to $[\exp^i(1)]^\mu \rightarrow [1]^\nu \rightarrow [\beta]^\pm$. Hence, $[\exp^i(1)]^\mu \rightarrow [\beta]^\nu \equiv_w [\exp^i(1)]^\mu \rightarrow [\beta]^\pm$ and we conclude by the first case. For $i = 0$ use Lemma 9.

B.3 Closure by \diamond

First of all note that since $\beta^+ \vee \beta^- = \beta^\pm$, by Lemma 8 $\alpha^\mu \diamond \beta^\pm$ corresponds to $(\alpha^\mu \diamond \beta^+) \vee (\alpha^\mu \diamond \beta^-)$. Therefore, if $\alpha^\mu \diamond \beta^\nu \in \Omega$, for both $\nu = +, -$, $\alpha^\mu \diamond \beta^\pm \in \Omega$ holds because we know that Ω is closed under \vee .

Lemma 10. Fix a natural number $n > 0$. For each $[\alpha]^\mu, [\beta]^\nu \in \hat{\Omega}_n$, one can effectively find $[\gamma_1]^{\lambda_1}, [\gamma_2]^{\lambda_2}, [\gamma_3]^{\lambda_3}$ such that $[\gamma_i]^{\lambda_i} \in \hat{\Omega}_n$ or $[\gamma_i]^{\lambda_i} = [\gamma'_i + 1]^{\lambda_i}$ and

$$[\alpha]^\mu \diamond [\beta]^\nu = [\gamma_1]^{\lambda_1}, \quad [\alpha + 1]^\mu \diamond [\beta]^\nu = [\gamma_2]^{\lambda_2}, \quad [\alpha + 1]^\mu \diamond [\beta + 1]^\nu = [\gamma_3]^{\lambda_3}.$$

Proof. We only give a proof of the first assertion. The remaining two are very similar.

We proceed by induction on the sum of the coefficients of α and β . The basic step is $\alpha = \exp^n(p)$, $\beta = \exp^n(q)$. By Lemma 6, we find r and λ such that $\alpha^\mu \diamond \beta^\nu = \exp^n(r)^\lambda$.

In the inductive step we give argument for the case $\alpha^\mu = \exp^n(p)^\mu \rightarrow (\gamma^- \vee \gamma^+)$ and $\beta^\nu = \exp^n(q)^\nu \rightarrow (\delta^- \vee \delta^+)$; the case when $\alpha = \exp^n(p)$ or $\beta = \exp^n(q)$ is very similar, only uses the second assertion of Lemma 7.

Since $\exp^n(p)^\mu$ and $\exp^n(q)^\nu$ are initializable, by Lemma 7 and Lemma 8, we get

$$\alpha^\mu \diamond \beta^\nu = \exp^n(p)^\mu \diamond \exp^n(q)^\nu \rightarrow (\alpha^\mu \diamond \delta^- \vee \alpha^\mu \diamond \delta^+ \vee \gamma^- \diamond \beta^\nu \vee \gamma^+ \diamond \beta^\nu).$$

By induction hypothesis we compute $\alpha^\mu \diamond \delta^-, \alpha^\mu \diamond \delta^+, \gamma^- \diamond \beta^\nu, \gamma^+ \diamond \beta^\nu \in \hat{\Omega}_n$ and by the properties of \vee we get $\eta^\kappa = \alpha^\mu \diamond \delta^- \vee \alpha^\mu \diamond \delta^+ \vee \gamma^- \diamond \beta^\nu \vee \gamma^+ \diamond \beta^\nu \in \hat{\Omega}_n$. Again by Lemma 6 we get an expression of the form $\exp^n(r)^\lambda \rightarrow \eta^\kappa$,

which can be presented as $\left(\sum_{i=N}^1 \exp^n(i)m_i\right)^{\lambda'}$. Indeed, by an argument analogous to the one used after Lemma 9, one shows that for $\eta = \sum_{i=M}^L \exp^n(i)\ell_i$ with $\ell_L > 0$ and for $\lambda \in \{+, -\}$,

$$[\exp^n(r)]^\lambda \rightarrow \eta^\kappa \equiv_w \begin{cases} \eta^\kappa & \text{if } L > r \text{ and } \kappa \in \{+, -\}, \text{ or} \\ & L = r \text{ and } \kappa = \lambda, \\ [\eta + \exp^n(r)]^\lambda & \text{if } L \geq r \text{ and } \kappa = \pm, \text{ or} \\ & L = r \text{ and } \kappa = \bar{\lambda}, \\ [\sum_{i=M}^r \exp^n(i)\ell_i + \exp^n(r)]^\lambda & \text{if } L < r. \end{cases}$$

It remains to show that $m_i \in \{0, 1\}$ for even i .

Suppose first that r is even, and $\lambda = -$. Then by Lemma 6 p and q are even, $r = p + q$ and $\mu = \nu = -$. For each of $\alpha^\mu \diamond \delta^-, \alpha^\mu \diamond \delta^+, \gamma^- \diamond \beta^\nu, \gamma^+ \diamond \beta^\nu$ we can apply Lemma 7, and get an expression $\exp^n(s)^\kappa \rightarrow \iota^{\kappa_1}$, where ι^{κ_1} can be again obtained from the inductive hypothesis. The possible values for $\exp^n(s)^\kappa$ are $\exp^n(p)^- \diamond \exp^n(t)^- = \exp^n(p+t)^-$ or $\exp^n(p)^- \diamond \exp^n(t)^+ = \exp^n(p+t)^+$ with $t > q$ (since q is even and $\beta^\nu \in \Omega$), or $\exp^n(q)^- \diamond \exp^n(u)^- = \exp^n(q+u)^-$ or $\exp^n(q)^- \diamond \exp^n(u)^+ = \exp^n(q+u)^+$ with $u > p$ (since p is even and $\alpha^\mu \in \Omega$). Hence, $s > p+q = r$. In consequence, by the properties of \rightarrow and \vee , $\eta^\kappa = \left(\sum_{i=L}^M \exp^n(i)\ell_i\right)^\kappa$ with $L > r, \ell_L > 0$. Hence,

$$\alpha^\mu \diamond \beta^\nu = \exp^n(r)^- \rightarrow \left(\sum_{i=L}^M \exp^n(i)\ell_i\right)^\kappa = \begin{cases} \left(\sum_{i=L}^M \exp^n(i)\ell_i\right)^\kappa + \exp^n(r)^- & \kappa = \pm \\ \left(\sum_{i=L}^M \exp^n(i)\ell_i\right)^\kappa & \kappa \in \{-, +\} \end{cases}.$$

In either case the result is in $\hat{\Omega}_n$.

Next, suppose that r is even, and $\lambda = +$. Then,

$$r = \begin{cases} p+q & p \equiv q \pmod{2}, \mu = \bar{\nu} \\ p+q-1 & p \text{ odd}, q \text{ even}, \nu = + \text{ (or symmetrically)} \\ p+q-2 & p, q \text{ even}, \mu = \nu = + \end{cases}$$

Again, we can present each of $\alpha^\mu \diamond \delta^-, \alpha^\mu \diamond \delta^+, \gamma^- \diamond \beta^\nu, \gamma^+ \diamond \beta^\nu$ as $\exp^n(t)^\kappa \rightarrow \iota^{\kappa_1}$ with $\iota^{\kappa_1} \in \hat{\Omega}_n$. In order to carry on like before it is enough to show that in no case $\exp^n(t)^\kappa = \exp^n(r)^-$. Indeed, if we exclude this possibility, by inductive hypothesis and by the properties of \rightarrow and \vee , we can conclude that

$$\eta^\kappa = \begin{cases} \left(\sum_{i=L}^M \exp^n(i)\ell_i\right)^\kappa \\ \left(\sum_{i=L}^M \exp^n(i)\ell_i\right)^\kappa + \exp^n(r)^+ \end{cases}$$

for $L > r, \ell_L > 0$, and so

$$\alpha^\mu \diamond \beta^\nu = \begin{cases} \left(\sum_{i=L}^M \exp^n(i)\ell_i\right)^\kappa \\ \left(\sum_{i=L}^M \exp^n(i)\ell_i\right)^\kappa + \exp^n(r)^+ \end{cases}.$$

These two are both in $\hat{\Omega}_n$, since η^κ is. So it remains to see that $\exp^n(t)^\kappa \neq \exp^n(r)^-$. Observe that only an expression of the form $\exp^n(2i)^- \diamond \exp^n(2j)^-$ can give $\exp^n(r)^-$ for even r . The only case giving a chance of such an expression is when $r = p+q$. But then $2i = p$ and, since $\beta^\nu \in \Omega$, $2j > q$ (or symmetrically), so $2i+2j > p+q = r$, and we are safe. If r is odd, the claim follows easily by induction hypothesis, and by properties of \rightarrow . \square

Lemma 11. Let $N > 0$ and $[\alpha_0]^\mu, [\beta_0]^\nu \in \hat{\Omega}_N$, with $\mu, \nu \in \{+, -\}$. Define $\hat{\alpha}_0 = \alpha_0 + \varepsilon_\alpha$, and $\hat{\beta}_0 = \beta_0 + \varepsilon_\beta$ with $\varepsilon_\alpha, \varepsilon_\beta \in \{0, 1\}$. Let $\alpha = \exp^N(\omega)(\omega^k a_k + \dots + \omega a_1 + a_0) + \hat{\alpha}_0$ and $\beta = \exp^N(\omega)(\omega^k b_k + \dots + \omega b_1 + b_0) + \hat{\beta}_0$. Let ℓ_α be the least i for which $a_i > 0$, and similarly for ℓ_β .

• If $\alpha_0 > 0, \beta_0 > 0$, then

$$[\alpha]^\mu \diamond [\beta]^\nu = [\hat{\alpha}_0]^\mu \diamond [\hat{\beta}_0]^\nu \rightarrow [\exp^N(\omega)(\omega^k(a_k + b_k) + \dots + \omega(a_1 + b_1) + (a_0 + b_0))]^\pm.$$

• If $\alpha_0 > 0, \hat{\beta}_0 = 0$, then

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^+ &= [\exp^N(\omega)(\omega^k(a_k + b_k) + \dots + \omega^{\ell_\beta}(a_{\ell_\beta} + b_{\ell_\beta}))]^+, \\ [\alpha]^\mu \diamond [\beta]^- &= [\hat{\alpha}_0]^\mu \rightarrow [\exp^N(\omega)(\omega^k(a_k + b_k) + \dots + \omega(a_1 + b_1) + (a_0 + b_0))]^\pm. \end{aligned}$$

• If $\alpha_0 = 0, \beta_0 = 0$, then

$$\begin{aligned} [\alpha]^- \diamond [\beta]^- &= [\exp^N(\omega)(\omega^k(a_k + b_k) + \dots + \omega(a_1 + b_1) + (a_0 + b_0))]^- && \text{for } \varepsilon_\alpha = \varepsilon_\beta = 0, \\ [\alpha]^- \diamond [\beta]^- &= [\exp^N(\omega)(\omega^k(a_k + b_k) + \dots + \omega(a_1 + b_1) + (a_0 + b_0) + 1)]^- && \text{for } \varepsilon_\alpha = \varepsilon_\beta = 1, \\ [\alpha]^- \diamond [\beta]^- &= [\exp^N(\omega)(\omega^k(a_k + b_k) + \dots + \omega(a_1 + b_1) + (a_0 + b_0) + 1)]^- && \text{for } \varepsilon_\alpha = 0, \varepsilon_\beta = 1, \ell_\beta \geq \ell_\alpha \\ [\alpha]^- \diamond [\beta]^- &= [\exp^N(\omega)(\omega^k(a_k + b_k) + \dots + \omega(a_1 + b_1) + (a_0 + b_0))]^- && \text{for } \varepsilon_\alpha = 0, \varepsilon_\beta = 1, \ell_\beta < \ell_\alpha \\ [\alpha]^- \diamond [\beta]^+ &= [\exp^N(\omega)(\omega^k(a_k + b_k) + \dots + \omega^{\ell_\beta}(a_{\ell_\beta} + b_{\ell_\beta}))]^+ && \text{for } \varepsilon_\beta = 0, \\ [\alpha]^- \diamond [\beta]^+ &= [\exp^N(\omega)(\omega^k(a_k + b_k) + \dots + \omega(a_1 + b_1) + (a_0 + b_0) + 1)]^+ && \text{for } \varepsilon_\alpha = 0, \varepsilon_\beta = 1, \\ [\alpha]^+ \diamond [\beta]^+ &= [\exp^N(\omega)(\omega^k(a_k + b_k) + \dots + \omega^\ell(a_\ell + b_\ell - \varepsilon))]^+ && \text{for } \varepsilon_\alpha = \varepsilon_\beta = 0, \\ [\alpha]^+ \diamond [\beta]^+ &= [\exp^N(\omega)(\omega^k(a_k + b_k) + \dots + \omega^{\ell_\beta}(a_{\ell_\beta} + b_{\ell_\beta}))]^+ && \text{for } \varepsilon_\alpha = 0, \varepsilon_\beta = 1, \end{aligned}$$

where $\ell = \max(\ell_\alpha, \ell_\beta)$ and ε equals 1 if $\ell_\alpha = \ell_\beta$ and 0 otherwise.

Proof. In the proof we assume that $\varepsilon_\alpha = \varepsilon_\beta = 0$, the remaining cases being very similar. We prove all the equations simultaneously by induction on (α, β) . Let $\theta = \exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i)$.

• Case $\alpha_0 > 0, \beta_0 > 0$. Suppose $\alpha = \alpha' + \exp^N(p), \beta = \beta' + \exp^N(q)$. We have

$$[\alpha]^\mu \diamond [\beta]^\nu = [\exp^N(p)]^\mu \diamond [\exp^N(q)]^\nu \rightarrow ([\alpha]^\mu \diamond [\beta']^\pm \vee [\alpha']^\pm \diamond [\beta]^\nu)$$

Let $\alpha_0 = \gamma + \exp^N(p), \beta_0 = \delta + \exp^N(p)$. We get

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^\nu &= [\exp^N(p)]^\mu \diamond [\exp^N(q)]^\nu \rightarrow (A \rightarrow [\theta]^\pm \vee B \rightarrow [\theta]^\pm) \\ &= [\exp^N(p)]^\mu \diamond [\exp^N(q)]^\nu \rightarrow (A \vee B) \rightarrow [\theta]^\pm \\ &= [\alpha_0]^\mu \diamond [\beta_0]^\nu \rightarrow [\theta]^\pm. \end{aligned}$$

with $A = [\alpha_0]^\mu \diamond [\delta]^\pm$ (or $A = [\alpha_0]^\mu$ if $\delta = 0$) and $B = [\gamma]^\pm \diamond [\beta_0]^\nu$ (or $B = [\beta_0]^\nu$ if $\gamma = 0$).

• *Case $\alpha_0 > 0, \beta_0 = 0$.* Suppose $\ell_\beta > 0$. Then by Proposition 6 we get that $[\exp^N(\omega)\omega^{\ell_\beta}]^+ = \sup_i^+ [\exp^N(\omega)\omega^{\ell_\beta-1}i]^+$. By the induction hypothesis for

$$b'_i = \begin{cases} b_i & i \neq \ell_\beta \\ b_i - 1 & i = \ell_\beta \end{cases}$$

we have

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^+ &= \sup_i^+ \alpha^\mu \diamond \left[\exp^N(\omega) \left(\left(\sum_{j=k}^{\ell_\beta} \omega^j b'_j \right) + \omega^{\ell_\beta-1} i \right) \right]^+ \\ &= \sup_i^+ \left[\exp^N(\omega) \left(\left(\sum_{j=k}^{\ell_\beta} \omega^j (a_j + b'_j) \right) + \omega^{\ell_\beta-1} (a_{\ell_\beta-1} + i) \right) \right]^+ \\ &= \left[\exp^N(\omega) \sum_{j=k}^{\ell_\beta} \omega^j (a_j + b_j) \right]^+. \end{aligned}$$

Now, suppose that $\ell_\beta = 0$ (this also covers the induction basis). Then $[\exp^N(\omega)]^+ = \sup_i^+ [\exp^N(i)]^+$, and like before, using the case $\alpha_0 > 0, \beta_0 > 0$,

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^+ &= \sup_i^+ \alpha^\mu \diamond \left[\exp^N(\omega) \left(\sum_{j=k}^{\ell_\beta} \omega^j b'_j \right) + \exp^N(i) \right]^+ \\ &= \sup_i^+ \left(\alpha_0^\mu \diamond [\exp^N(i)]^+ \rightarrow \left[\exp^N(\omega) \sum_{j=k}^0 \omega^j (a_j + b'_j) \right]^\pm \right) \\ &= [\exp^N(\omega)]^+ \rightarrow \left[\exp^N(\omega) \left(\sum_{j=k}^0 \omega^j (a_j + b'_j) \right) \right]^\pm \\ &= \left[\exp^N(\omega) \sum_{j=k}^{\ell_\beta} \omega^j (a_j + b_j) \right]^+. \end{aligned}$$

Let us move to the second equation. Let $\alpha = \alpha' + \exp^N(p)$, $\beta = \beta' + \exp^N(\omega)\omega^{\ell_\beta}$. If $\ell_\beta > 0$, we have

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^- &= [\exp^N(p)]^\mu \rightarrow \left(([\alpha']^\pm \diamond [\beta]^-) \vee (\sup_n [\alpha]^\mu \diamond [\beta' + \exp^N(\omega)\omega^{\ell_\beta-1}n]^+) \vee \right. \\ &\quad \left. \vee (\sup_n^+ [\alpha]^\mu \diamond [\beta' + \exp^N(\omega)\omega^{\ell_\beta-1}n]^+) \right). \end{aligned}$$

As $[\alpha]^\mu \diamond [\beta' + \exp^N(\omega)\omega^{\ell_\beta-1}n]^+ = \left[\exp^N(\omega) \left(\left(\sum_{i=k}^{\ell_\beta} \omega^i (a_i + b'_i) \right) + \omega^{\ell_\beta-1} (a_{\ell_\beta-1} + n) \right) \right]^+$, we obtain

$$[\alpha]^\mu \diamond [\beta]^- = [\exp^N(p)]^\mu \rightarrow \left(([\alpha']^\pm \diamond [\beta]^-) \vee \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^\pm \right).$$

If $\alpha' = \exp^N(\omega)\gamma + \delta$, with $[\delta]^\pm \in \hat{\Omega}_N$, $\delta > 0$, then we conclude from the induction hypothesis that

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^- &= [\exp^N(p)]^\mu \rightarrow \left(([\delta]^\pm \rightarrow [\theta]^\pm) \vee \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^\pm \right) \\ &= [\exp^N(p)]^\mu \rightarrow ([\delta]^\pm \rightarrow [\theta]^\pm) \\ &= ([\exp^N(p)]^\mu \rightarrow [\delta]^\pm) \rightarrow [\theta]^\pm \\ &= [\alpha_0]^\mu \rightarrow [\theta]^\pm. \end{aligned}$$

If $\alpha' = \exp^N(\omega)\gamma$, by induction hypothesis $[\alpha']^\pm \diamond [\beta]^- = [\theta]^- \vee \left[\exp^N(\omega) \sum_{i=k}^{\ell_\alpha} \omega^i (a_i + b_i) \right]^+$, so

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^- &= [\exp^N(p)]^\mu \rightarrow \left([\theta]^- \vee \left[\exp^N(\omega) \sum_{i=k}^{\ell_\alpha} \omega^i (a_i + b_i) \right]^+ \vee \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^\pm \right) \\ &= [\exp^N(p)]^\mu \rightarrow [\theta]^\pm \end{aligned}$$

as one of the last two disjuncts must be at least $[\theta]^+$. This concludes the case of $\ell_\beta > 0$.

If $\ell_\beta = 0$ we have

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^- &= [\exp^N(p)]^\mu \rightarrow \left(([\alpha']^\pm \diamond [\beta]^-) \vee (\sup_n [\alpha]^\mu \diamond [\beta' + \exp^N(n)]^+) \vee \right. \\ &\quad \left. \vee (\sup_n^+ [\alpha]^\mu \diamond [\beta' + \exp^N(n)]^+) \right). \end{aligned}$$

Since $[\alpha]^\mu \diamond [\beta' + \exp^N(n)]^+ = [\alpha_0]^\mu \diamond [\exp^N(n)]^+ \rightarrow \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b'_i) \right]^\pm$, by Lemma 10 we have

$$[\alpha]^\mu \diamond [\beta]^- = [\exp^N(p)]^\mu \rightarrow \left(([\alpha']^\pm \diamond [\beta]^-) \vee [\theta]^\pm \right)$$

If $\alpha' = \exp^N(\omega)\gamma + \delta$, then $\alpha_0 = \delta + \exp^N(p)$ and by induction hypothesis for $[\alpha']^\pm \diamond [\beta]^-$ we get

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^- &= [\exp^N(p)]^\mu \rightarrow \left(([\delta]^\pm \rightarrow [\theta]^\pm) \vee [\theta]^\pm \right) \\ &= [\exp^N(p)]^\mu \rightarrow [\delta]^\pm \rightarrow [\theta]^\pm \\ &= [\alpha_0]^\mu \rightarrow [\theta]^\pm. \end{aligned}$$

If $\alpha' = \exp^N(\omega)\gamma$, then $\alpha_0 = \exp^N(p)$ and by induction hypothesis $[\alpha']^\pm \diamond [\beta]^-$ is at most $[\theta]^\pm$. Hence,

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^- &= [\exp^N(p)]^\mu \rightarrow [\theta]^\pm \\ &= [\alpha_0]^\mu \rightarrow [\theta]^\pm. \end{aligned}$$

• *Case* $\alpha_0 = 0, \beta_0 = 0$. Let $\alpha = \alpha' + \exp^N(\omega)\omega^{\ell_\alpha}, \beta = \beta' + \exp^N(\omega)\omega^{\ell_\beta}$. Suppose that $\ell_\alpha > 0, \ell_\beta > 0$. Set $a_{\ell_\alpha}^{(n)} = a_{\ell_\alpha} - 1, a_{\ell_\alpha-1}^{(n)} = n$, and $a_i^{(n)} = a_i$ for other i 's. Analogously define $b_i^{(n)}$. Applying Lemma 7 and the inductive hypothesis we get the claim

$$\begin{aligned}
& [\alpha]^- \diamond [\beta]^- = \\
& = \perp \rightarrow \left(\bar{\sup}_n ([\alpha]^- \diamond [\beta' + \exp^N(\omega)\omega^{\ell_\beta-1}n]^-) \vee \bar{\sup}_n ([\alpha' + \exp^N(\omega)\omega^{\ell_\alpha-1}n]^- \diamond [\beta]^-) \right) \\
& = \perp \rightarrow \left(\bar{\sup}_n \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i^{(n)}) \right]^- \vee \bar{\sup}_n \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i^{(n)} + b_i) \right]^- \right) \\
& = \perp \rightarrow \left(\left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^- \vee \left[\exp^N(\omega) \sum_{i=k}^{\ell_\alpha} \omega^i (a_i + b_i) \right]^- \right) \\
& = \perp \rightarrow \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i) \right]^- = \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i) \right]^- , \\
& [\alpha]^- \diamond [\beta]^+ = \\
& = \top \rightarrow \left(\bar{\sup}_n^+ ([\alpha]^- \diamond [\beta' + \exp^N(\omega)\omega^{\ell_\beta-1}n]^-) \vee \bar{\sup}_n^+ ([\alpha' + \exp^N(\omega)\omega^{\ell_\alpha-1}n]^- \diamond [\beta]^+) \right) \\
& = \top \rightarrow \left(\bar{\sup}_n^+ \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i^{(n)}) \right]^- \vee \bar{\sup}_n^+ \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i^{(n)} + b_i) \right]^- \right) \\
& = \top \rightarrow \left(\left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^+ \vee \left[\exp^N(\omega) \sum_{i=k}^{\max(\ell_\alpha, \ell_\beta)} \omega^i (a_i + b_i) \right]^+ \right) \\
& = \top \rightarrow \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^+ = \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^+ , \\
& [\alpha]^+ \diamond [\beta]^+ = \\
& = \top \rightarrow \left(\bar{\sup}_n^+ ([\alpha]^+ \diamond [\beta' + \exp^N(\omega)\omega^{\ell_\beta-1}n]^-) \vee \bar{\sup}_n^+ ([\alpha' + \exp^N(\omega)\omega^{\ell_\alpha-1}n]^- \diamond [\beta]^+) \right) \\
& = \top \rightarrow \left(\bar{\sup}_n^+ \left[\exp^N(\omega) \sum_{i=k}^{\ell_\alpha} \omega^i (a_i + b_i^{(n)}) \right]^- \vee \bar{\sup}_n^+ \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i^{(n)} + b_i) \right]^- \right) \\
& = \top \rightarrow \left(\left[\exp^N(\omega) \sum_{i=k}^{\max(\ell_\alpha, \ell_\beta)} \omega^i (a_i + b_i) \right]^+ \vee \left[\exp^N(\omega) \sum_{i=k}^{\max(\ell_\alpha, \ell_\beta)} \omega^i (a_i + b_i) \right]^+ \right) \\
& = \top \rightarrow \left[\exp^N(\omega) \sum_{i=k}^{\max(\ell_\alpha, \ell_\beta)} \omega^i (a_i + b_i) \right]^+ = \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^+ .
\end{aligned}$$

Next suppose that $\ell_\alpha = 0, \ell_\beta > 0$. Similarly,

$$\begin{aligned}
& [\alpha]^- \diamond [\beta]^- = \\
& = \perp \rightarrow \left(\bar{\sup}_n ([\alpha]^- \diamond [\beta' + \exp^N(\omega)\omega^{\ell_\beta-1}n]^-) \vee \bar{\sup}_n ([\alpha' + \exp^N(n)]^- \diamond [\beta]^-) \right) \\
& = \perp \rightarrow \left(\bar{\sup}_n \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i^{(n)}) \right]^- \vee \bar{\sup}_n \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a'_i + b_i) + \exp^N(n) \right]^- \right) \\
& = \perp \rightarrow \left(\left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^- \vee \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i) \right]^- \right) \\
& = \perp \rightarrow \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i) \right]^- = \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i) \right]^- , \\
& [\alpha]^- \diamond [\beta]^+ = \\
& = \top \rightarrow \left(\bar{\sup}_n^+ ([\alpha]^- \diamond [\beta' + \exp^N(\omega)\omega^{\ell_\beta-1}n]^-) \vee \bar{\sup}_n^+ ([\alpha' + \exp^N(n)]^- \diamond [\beta]^+) \right) \\
& = \top \rightarrow \left(\bar{\sup}_n^+ \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i^{(n)}) \right]^- \vee \bar{\sup}_n^+ \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a'_i + b_i) \right]^- \right) \\
& = \top \rightarrow \left(\left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^+ \vee \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a'_i + b_i) \right]^+ \right) \\
& = \top \rightarrow \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^+ = \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^+ , \\
& [\alpha]^+ \diamond [\beta]^- = \\
& = \top \rightarrow \left(\bar{\sup}_n^+ ([\alpha]^+ \diamond [\beta' + \exp^N(\omega)\omega^{\ell_\beta-1}n]^-) \vee \bar{\sup}_n^+ ([\alpha' + \exp^N(n)]^- \diamond [\beta]^-) \right) \\
& = \top \rightarrow \left(\bar{\sup}_n^+ \left[\exp^N(\omega) \sum_{i=k}^{\ell_\alpha} \omega^i (a_i + b_i^{(n)}) \right]^- \vee \bar{\sup}_n^+ \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a'_i + b_i) + \exp^N(n) \right]^- \right) \\
& = \top \rightarrow \left(\left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^+ \vee \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i) \right]^+ \right) \\
& = \top \rightarrow \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i) \right]^+ = \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b_i) \right]^+ = \left[\exp^N(\omega) \sum_{i=k}^{\ell_\alpha} \omega^i (a_i + b_i) \right]^+ ,
\end{aligned}$$

$$\begin{aligned}
[\alpha]^+ \diamond [\beta]^+ &= \\
&= \top \rightarrow \left(\overset{\dagger}{\sup}_n ([\alpha]^+ \diamond [\beta' + \exp^N(\omega)\omega^{\ell_\beta-1}n]^-) \vee \overset{\dagger}{\sup}_n ([\alpha' + \exp^N(n)]^- \diamond [\beta]^+) \right) \\
&= \top \rightarrow \left(\overset{\dagger}{\sup}_n \left[\exp^N(\omega) \sum_{i=k}^{\ell_\alpha} \omega^i (a_i + b_i^{(n)}) \right]^- \vee \overset{\dagger}{\sup}_n \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a'_i + b_i) \right]^- \right) \\
&= \top \rightarrow \left(\left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^+ \vee \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a'_i + b_i) \right]^+ \right) \\
&= \top \rightarrow \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^+ = \left[\exp^N(\omega) \sum_{i=k}^{\ell_\beta} \omega^i (a_i + b_i) \right]^+.
\end{aligned}$$

Finally, for $\ell_\alpha = 0, \ell_\beta = 0$

$$\begin{aligned}
[\alpha]^- \diamond [\beta]^- &= \\
&= \perp \rightarrow \left(\overset{\dagger}{\sup}_n ([\alpha]^- \diamond [\beta' + \exp^N(n)]^-) \vee \overset{\dagger}{\sup}_n ([\alpha' + \exp^N(n)]^- \diamond [\beta]^-) \right) \\
&= \perp \rightarrow ([\theta]^- \vee [\theta]^-) = \perp \rightarrow [\theta]^- = [\theta]^-,
\end{aligned}$$

$$\begin{aligned}
[\alpha]^- \diamond [\beta]^+ &= \\
&= \top \rightarrow \left(\overset{\dagger}{\sup}_n ([\alpha]^- \diamond [\beta' + \exp^N(n)]^-) \vee \overset{\dagger}{\sup}_n ([\alpha' + \exp^N(n)]^- \diamond [\beta]^+) \right) \\
&= \top \rightarrow \left([\theta]^+ \vee \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a'_i + b_i) \right]^+ \right) = \top \rightarrow [\theta]^+ = [\theta]^+,
\end{aligned}$$

$$\begin{aligned}
[\alpha]^+ \diamond [\beta]^+ &= \\
&= \top \rightarrow \left(\overset{\dagger}{\sup}_n ([\alpha]^+ \diamond [\beta' + \exp^N(n)]^-) \vee \overset{\dagger}{\sup}_n ([\alpha' + \exp^N(n)]^- \diamond [\beta]^+) \right) \\
&= \top \rightarrow \left(\left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a_i + b'_i) \right]^+ \vee \left[\exp^N(\omega) \sum_{i=k}^0 \omega^i (a'_i + b_i) \right]^+ \right),
\end{aligned}$$

and since $\sum_{i=k}^0 \omega^i (a_i + b'_i) = \sum_{i=k}^0 \omega^i (a'_i + b_i) = \left(\sum_{i=k}^1 \omega^i (a_i + b_i) \right) + (a_0 + b_0 - 1)$, the claim follows. \square

Note that $[0]^\mu$ is not a valid set. Nevertheless, to simplify notation we adopt a convention $[\alpha]^\mu \diamond [0]^\nu = [\alpha]^\mu$.

Lemma 12. Let $[\alpha]^\mu, [\beta]^\nu \in \Omega$ and $\alpha = \sum_{i=N}^K \alpha_i, \beta = \sum_{j=N}^L \beta_j, \alpha_i \in \Omega_i, \beta_j \in \Omega_j, \alpha_K, \beta_L > 0, \mu, \nu \in \{+, -\}$. For $J < \omega$ let $\alpha_{J<} = \sum_{i=N}^{J+1} \alpha_i, \beta_{J<} = \sum_{i=N}^{J+1} \beta_i$.

• If $K < L$ and $[\beta_L]^\nu = [\exp^L(\omega)\gamma]^-$

$$[\alpha]^\mu \diamond [\beta]^\nu = \begin{cases} \left[\sum_{i=L-1}^K \alpha_i \right]^\mu \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\pm & \text{for } \sum_{i=L-1}^K \alpha_i \geq \omega, \\ \left[\sum_{i=L-1}^K \alpha_i - 1 \right]^\mu \rightarrow [\alpha_{L-1<} + 1]^\pm \diamond [\beta]^\pm & \text{for } \omega > \sum_{i=L-1}^K \alpha_i > 1, \\ [1]^+ \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu & \text{for } \left[\sum_{i=L-1}^K \alpha_i \right]^\mu = [1]^+, \\ [\alpha_L + 1]^- \diamond [\beta_L]^\nu \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^- & \text{for } \left[\sum_{i=L-1}^K \alpha_i \right]^\mu = [1]^-. \end{cases}$$

• If $K < L$ and $[\beta_L]^\nu \neq [\exp^L(\omega)\gamma]^-$

$$[\alpha]^\mu \diamond [\beta]^\nu = \begin{cases} [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu & \text{for } \sum_{i=L-1}^K \alpha_i > 1, \\ [1]^+ \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu & \text{for } \left[\sum_{i=L-1}^K \alpha_i \right]^\mu = [1]^+, \\ [\alpha_L + 1]^- \diamond [\beta_L]^\nu \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^- & \text{for } \left[\sum_{i=L-1}^K \alpha_i \right]^\mu = [1]^-. \end{cases}$$

• If $K = L$

$$[\alpha]^\mu \diamond [\beta]^\nu = \begin{cases} [\alpha_L]^\mu \diamond [\beta_L]^\nu \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^- & \text{for } \alpha_L, \beta_L \geq \omega, \\ [\alpha_L]^\mu \diamond [\beta_L - 1]^\nu \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^\pm & \text{for } \alpha_L \geq \omega > \beta_L > 1, \\ [\alpha_L - 1]^\mu \diamond [\beta_L - 1]^\nu \rightarrow [\alpha_{L<} + 1]^\pm \diamond [\beta_{L<} + 1]^\pm & \text{for } \omega > \alpha_L \cdot \beta_L > 1, \\ [\alpha_M + 1]^\mu \diamond [\beta_M + 1]^\nu \rightarrow [\alpha_{M<} + 1]^- \diamond [\beta_{M<} + 1]^- & \text{for } \alpha_L \cdot \beta_L = 1, \end{cases}$$

where M is the least $i > 1$ such that α_i or β_i is nonzero.

Proof. Again, we prove all the equations simultaneously by induction on (α, β) . It is easy to observe that the inequality “ \geq ” is in each case very simple. We concentrate on “ \leq ”. Further more, since in all three cases the arguments are very similar we only give the calculations for the first case: $K < L$ and $[\beta_L]^\nu = [\exp^L(\omega)\gamma]^-$.

Suppose first that $\alpha_K = \alpha'_K + \exp^K(\omega)\omega^p, K > 0$. Let $\alpha' = \alpha_{K<} + \alpha'_K, \beta' = \beta_{K<} + \beta'_K$. We only give the calculation for $p, q > 0$ (if one or both are zero the calculation is entirely analogous):

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^\nu &= 1^\mu \rightarrow (\sup_m^\mu [\alpha' + \exp^K(\omega)\omega^{p-1}m]^- \diamond [\beta]^-) \vee (\sup_n^\mu [\alpha]^\mu \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^+) \\ &= 1^\mu \rightarrow \left(\sup_m^\mu \left[\left(\sum_{i=L-1}^{K+1} \alpha_i \right) + \alpha'_K + \exp^K(\omega)\omega^{p-1}m \right]^- \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\pm \right) \vee \\ &\quad \vee \left(\sup_n^\mu [\alpha_{L-1<} + 1]^- \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^+ \right) \\ &\leq 1^\mu \rightarrow \left(\left[\sum_{i=L-1}^K \alpha_i \right]^\mu \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\pm \right) \vee ([\alpha_{L-1<} + 1]^- \diamond [\beta]^\mu) \\ &= \left[\sum_{i=L-1}^K \alpha_i \right]^\mu \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\pm. \end{aligned}$$

Suppose now that $\alpha_K = \alpha'_K + \exp^K(p)$, $K > 0$. For $q > 0$

$$\begin{aligned}
[\alpha]^\mu \diamond [\beta]^\nu &= [\exp^K(p)]^\mu \rightarrow [\alpha']^\pm \diamond [\beta]^- \vee \sup_n^+ [\alpha]^\mu \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^+ \vee \\
&\quad \vee \sup_n \bar{[\alpha]^\mu \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^+} \\
&\leq [\exp^K(p)]^\mu \rightarrow \left(\left[\left(\sum_{i=L-1}^{K+1} \alpha_i \right) + \alpha'_K \right]^\pm \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\pm \right) \vee \\
&\quad \vee ([\alpha_{L-1<} + 1]^- \diamond [\beta]^+) \vee ([\alpha_{L-1<} + 1]^- \diamond [\beta]^-) \\
&\leq \left[\sum_{i=L-1}^K \alpha_i \right]^\mu \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\pm,
\end{aligned}$$

and for $q = 0$ the computation is analogous.

Finally suppose that $K = 0$. For $\alpha_K = \alpha'_K + \omega^p$ and $\alpha_K = \alpha'_K + 1$ the calculations are entirely analogous to the ones above save for three cases we consider below. For $\sum_{i=L-1}^K \alpha_i = \omega$

$$\begin{aligned}
[\alpha]^\mu \diamond [\beta]^\nu &= 1^\mu \rightarrow (\sup_m^\mu [\alpha' + m]^- \diamond [\beta]^-) \vee (\sup_n^\mu [\alpha]^\mu \diamond [\beta'_L + \exp^L(\omega)\omega^{q-1}n]^+) \\
&\leq 1^\mu \rightarrow \left(\sup_m^\mu [m-1]^- \rightarrow [\alpha_{L-1<} + 1]^\pm \diamond [\beta]^\pm \right) \vee ([\alpha_{L-1<} + 1]^- \diamond [\beta]^\mu) \\
&\leq 1^\mu \rightarrow \left(\sup_m^\mu [m-1]^- \rightarrow 1^+ \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\pm \right) \vee ([\alpha_{L-1<} + 1]^- \diamond [\beta]^\mu) \\
&\leq [\omega]^\mu \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\pm,
\end{aligned}$$

since $[\alpha_{L-1<} + 1]^+ \diamond [\beta]^\pm \leq \top \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^- \leq \top \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\pm$.

For $\sum_{i=L-1}^K \alpha_i = 2$

$$\begin{aligned}
[\alpha]^\mu \diamond [\beta]^\nu &\leq [1]^\mu \rightarrow [\alpha']^\pm \diamond [\beta]^- \vee \sup_n^\mu [\alpha]^\mu \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^+ \\
&\leq [1]^\mu \rightarrow ([\alpha_{L-1<} + 1]^\pm \diamond [\beta]^\mu) \vee \\
&\quad \vee \sup_n^\mu ([1]^\mu \rightarrow [\alpha_{L-1<} + 1]^\pm \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^\pm) \\
&\leq [1]^\mu \rightarrow ([\alpha_{L-1<} + 1]^\pm \diamond [\beta]^\mu) \vee \sup_n^\mu ([1]^\mu \rightarrow [\alpha_{L-1<} + 1]^\pm \diamond [\beta]^\nu) \\
&\leq [1]^\mu \rightarrow ([\alpha_{L-1<} + 1]^\pm \diamond [\beta]^\mu) \vee ([1]^\mu \rightarrow [1]^\mu \rightarrow [\alpha_{L-1<} + 1]^\pm \diamond [\beta]^\nu) \\
&= [1]^\mu \rightarrow [\alpha_{L-1<} + 1]^\pm \diamond [\beta]^\pm.
\end{aligned}$$

For $\left[\sum_{i=L-1}^K \alpha_i\right]^\mu = [1]^+$

$$\begin{aligned}
[\alpha]^\mu \diamond [\beta]^\nu &= [1]^+ \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta]^- \vee \left(\sup_n^+ [\alpha_{L-1<} + 1]^+ \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^- \right) \\
&\leq [1]^\mu \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta]^- \vee \left(\sup_n^+ [1]^+ \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^- \right) \\
&\leq [1]^\mu \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta]^- \vee \left(\sup_n^+ [1]^+ \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta]^- \right) \\
&= [1]^\mu \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta]^- \vee ([1]^+ \rightarrow [1]^+ \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta]^-) \\
&= [1]^\mu \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta]^-
\end{aligned}$$

For $\left[\sum_{i=L-1}^K \alpha_i\right]^\mu = [1]^-$

$$\begin{aligned}
[\alpha]^\mu \diamond [\beta]^\nu &= [1]^- \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta]^- \vee \left(\sup_n^- [\alpha_{L-1<} + 1]^- \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^- \right) \\
&\leq [1]^- \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta]^- \vee \\
&\quad \vee \left(\sup_n^- [\alpha_L + 1]^- \diamond [\beta'_L + \exp^L(\omega)\omega^{q-1}n]^- \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^- \right) \\
&\leq [1]^- \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta]^- \vee ([\alpha_L + 1]^- \diamond [\beta_L]^- \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^-)
\end{aligned}$$

If $\alpha_L > 0$ we get $[\alpha_{L-1<}]^\pm \diamond [\beta]^- = [\alpha_L]^\pm \diamond [\beta_L]^- \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^-$. If $\alpha_L = 0$, let N be the least $i > L$ such that $\alpha_i > 0$. Then we get $[\alpha_{L-1<}]^- \diamond [\beta]^- = [\alpha_{N-1<}]^- \diamond [\beta]^- \leq \left[\sum_{i=N-1}^L \beta_i\right]^- \rightarrow [\alpha_{N-1<}]^\pm \diamond [\beta_{N-1<} + 1]^- = [\beta_L]^- \rightarrow [\alpha_{N-1<}]^\pm \diamond [\beta_{L<}]^- \leq [\beta_L]^- \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^-$. In either case we have

$$\begin{aligned}
[\alpha]^\mu \diamond [\beta]^\nu &\leq [1]^- \rightarrow ([\alpha_L + 1]^- \diamond [\beta_L]^- \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^-) \\
&\leq [\alpha_L + 1]^- \diamond [\beta_L]^- \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^- .
\end{aligned}$$

Case $K < L$ and $[\beta_L]^\nu = [\beta'_L + \exp^L(\omega)\omega^q]^+$. Suppose first that $\alpha_K = \alpha'_K + \exp^K(\omega)\omega^p$. Let $\alpha' = \alpha_{K<} + \alpha'_K$, $\beta' = \beta_{K<} + \beta'_K$. We can treat $K > 0$ and $K = 0$ uniformly. We only give the calculation for $p, q > 0$ (if one or both are zero its entirely analogous):

$$\begin{aligned}
[\alpha]^\mu \diamond [\beta]^\nu &= [1]^+ \rightarrow \left(\sup_m^+ [\alpha' + \exp^K(\omega)\omega^{p-1}m]^+ \diamond [\beta]^+ \right) \vee \left(\sup_n^+ [\alpha]^\mu \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^+ \right) \\
&= [1]^+ \rightarrow \left(\sup_m^+ [\alpha_{L-1<} + 1]^- \diamond [\beta]^+ \right) \vee \left(\sup_n^+ [\alpha_{L-1<} + 1]^- \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^+ \right) \\
&= [1]^+ \rightarrow ([1]^+ \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^+) \vee ([\alpha_{L-1<} + 1]^- \diamond [\beta]^+) \\
&= [1]^+ \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^+ = [\alpha_{L-1<} + 1]^- \diamond [\beta]^+ .
\end{aligned}$$

Suppose now that $\alpha_K = \alpha'_K + \exp^K(p)$ or $\alpha_K = \alpha'_K + p$ with $\sum_{i=L-1}^K \alpha_i > 1$. For $q > 0$

$$\begin{aligned}
[\alpha]^\mu \diamond [\beta]^\nu &= \sup_n^+ [\alpha]^\mu \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^+ \\
&= \sup_n^+ [\alpha' + 1]^- \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^+ \\
&= [\alpha_{L-1<} + 1]^- \diamond [\beta]^+
\end{aligned}$$

and for $q = 0$ the computation is analogous.

For $\left[\sum_{i=L-1}^K \alpha_i\right]^\mu = [1]^+$ we have

$$\begin{aligned}
[\alpha]^\mu \diamond [\beta]^\nu &= \sup_n^+ [\alpha]^\mu \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^+ \\
&= \sup_n^+ [1]^+ \rightarrow [\alpha_{L-1<}^\pm]^\pm \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^+ \\
&\leq \sup_n^+ [1]^+ \rightarrow [\alpha_{L-1<}^\pm]^\pm \diamond [\beta]^+ \\
&= [1]^+ \rightarrow [1]^+ \rightarrow [\alpha_{L-1<}^\pm]^\pm \diamond [\beta]^+ \\
&\leq [1]^+ \rightarrow [\alpha_{L-1<}^\pm]^\pm \diamond [\beta]^+.
\end{aligned}$$

For $\left[\sum_{i=L-1}^K \alpha_i\right]^\mu = [1]^-$

$$\begin{aligned}
[\alpha]^\mu \diamond [\beta]^\nu &= \sup_n^+ [\alpha]^\mu \diamond [\beta' + \exp^L(\omega)\omega^{q-1}n]^+ \\
&= \sup_n^+ [\alpha_L + 1]^- \diamond [\beta'_L + \exp^L(\omega)\omega^{q-1}n]^+ \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^- \\
&= [\alpha_L + 1]^- \diamond [\beta_L]^+ \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^- .
\end{aligned}$$

Case $K < L$ and $[\beta_L]^\nu = [\beta'_L + \exp^L(q)]^\nu$. Suppose first that $\alpha_K = \alpha'_K + \exp^K(\omega)\omega^p$. Let $\alpha' = \alpha_{K<} + \alpha'_K$, $\beta' = \beta_{K<} + \beta'_K$. We can treat $K > 0$ and $K = 0$ uniformly. We only give the calculation for $p > 0$ (if $p = 0$ it is entirely analogous). For $\mu = -$ we have

$$\begin{aligned}
[\alpha]^\mu \diamond [\beta]^\nu &= [\exp^L(q)]^\nu \rightarrow [\alpha]^- \diamond [\beta']^\pm \vee (\sup_m^+ [\alpha' + \exp^K(\omega)\omega^{p-1}m]^+ \diamond [\beta]^\nu) \vee (\sup_n^- [\alpha' + \exp^K(\omega)\omega^{p-1}m]^+ \diamond [\beta]^\nu) \\
&= [\exp^L(q)]^\nu \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta']^\pm \vee (\sup_m^+ [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu) \vee (\sup_n^- [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu) \\
&= [\exp^L(q)]^\nu \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta']^\pm \vee ([1]^+ \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu) \vee ([1]^- \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu) \\
&= [\exp^L(q)]^\nu \rightarrow [1]^\pm \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu \\
&= [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu ,
\end{aligned}$$

and for $\mu = +$

$$\begin{aligned}
[\alpha]^\mu \diamond [\beta]^\nu &= \sup_m^+ [\alpha' + \exp^K(\omega)\omega^{p-1}m]^+ \diamond [\beta]^\nu \\
&= \sup_m^+ [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu \\
&= [1]^+ \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu \\
&= [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu .
\end{aligned}$$

Suppose now that $\alpha_K = \alpha'_K + \exp^K(p)$. We have

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^\nu &= [\exp^K(p)]^\mu \diamond [\exp^L(q)]^\nu \rightarrow [\alpha]^\mu \diamond [\beta']^\pm \vee [\alpha']^\pm \diamond [\beta]^\nu \\ &= [\exp^L(q)]^\nu \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta']^\pm \vee [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu \\ &= [\exp^L(q)]^\nu \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu \\ &= [\alpha_{L-1<} + 1]^- \diamond [\beta]^\nu. \end{aligned}$$

For $\alpha_K = \alpha'_K + p$ with $\sum_{i=L-1}^K \alpha_i > 1$ the computation is just like above. For $\left[\sum_{i=L-1}^K \alpha_i\right]^\mu = [1]^+$ we have

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^\nu &= [1]^+ \diamond [\exp^L(q)]^\nu \rightarrow [\alpha]^\mu \diamond [\beta']^\pm \vee [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu \\ &= [1]^+ \rightarrow [\alpha_{L-1<} + 1]^+ \diamond [\beta']^\pm \vee [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu \end{aligned}$$

If $\beta'_L > 0$, we can transform this further as

$$\begin{aligned} &= [1]^+ \rightarrow (([1]^+ \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta']^\pm) \vee [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu) \\ &= [1]^+ \rightarrow (([\alpha_{L-1<}]^\pm \diamond [\beta']^\pm) \vee [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu) \\ &= [1]^+ \rightarrow ([\alpha_{L-1<}]^\pm \diamond [\beta]^\nu) \end{aligned}$$

and we are done. Suppose $\beta'_L = 0$. Let M be the least $i > L$ such that $\beta_i > 0$. If $\beta_M = \beta'_M + \exp^M(\omega)\omega^r$, then we have

$$\begin{aligned} &= [1]^+ \rightarrow \left(\left[\sum_{i=M-1}^{L_1} \alpha_i + 1 \right]^+ \rightarrow [\alpha_{M-1<}]^\pm \diamond [\beta']^\pm \right) \vee [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu \\ &= [1]^+ \rightarrow (([1]^+ \rightarrow [\alpha_{L-1<}]^\pm \diamond [\beta']^\pm) \vee [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu) \\ &= [1]^+ \rightarrow (([\alpha_{L-1<}]^\pm \diamond [\beta']^\pm) \vee [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu) \\ &= [1]^+ \rightarrow ([\alpha_{L-1<}]^\pm \diamond [\beta]^\nu). \end{aligned}$$

If $\beta_M = \beta'_M + \exp^M(r)$, then we have

$$\begin{aligned} &= [1]^+ \rightarrow (([\alpha_{M-1<}]^\pm \diamond [\beta']^\pm) \vee [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu) \\ &= [1]^+ \rightarrow (([\alpha_{L-1<}]^\pm \diamond [\beta']^\pm) \vee [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu) \\ &= [1]^+ \rightarrow ([\alpha_{L-1<}]^\pm \diamond [\beta]^\nu). \end{aligned}$$

For $\left[\sum_{i=L-1}^K \alpha_i\right]^\mu = [1]^-$

$$\begin{aligned} [\alpha]^\mu \diamond [\beta]^\nu &= [1]^- \diamond [\exp^L(q)]^\nu \rightarrow [\alpha]^\mu \diamond [\beta']^\pm \vee [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu \\ &= [\exp^L(q)]^\nu \rightarrow [\alpha_{L-1<} + 1]^- \diamond [\beta']^\pm \vee [\alpha_{L-1<}]^\pm \diamond [\beta]^\nu \\ &= [\beta_L]^\nu \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^- \\ &= [\alpha_L + 1]^- \diamond [\beta_L]^\nu \rightarrow [\alpha_{L<} + 1]^- \diamond [\beta_{L<} + 1]^- \end{aligned}$$

Case $K = L$. Using similar computations as above, follow the subcases of Lemma 11. \square

Lemma 12 gives a recursive procedure to compute the result of $[\alpha]^\mu \diamond [\beta]^\nu$. Observe that in each case we reduce the problem to at most two instances: one on elements of (roughly speaking) the same Ω_N , the other on sets corresponding to ordinals such that either their respective K and L are smaller, or are obtained by cutting off a nontrivial (bigger than 1) tail part of the sum defining the ordinal, and replacing it with 1. The first subproblem is solved by Lemma 11, and the second problem solved recursively.

What remains to be done, is put the results of the two subcomputations together. For this we need to prove that the class Ω is closed by operations of the form $[\sum_{i=L}^0 \alpha_i]^\mu \rightarrow [\sum_{i=M}^{L+1} \alpha_i + j]^\nu$ with $j < \omega$, and $\alpha_i \in \Omega_i$.

Clearly, $[\sum_{i=L}^0 \alpha_i]^\mu \rightarrow [\sum_{i=M}^{L+1} \alpha_i + j]^\nu = [\sum_{i=L}^0 \alpha_i]^\mu \rightarrow [\sum_{i=M}^{L+1} \alpha_i]^\pm$, if $j > 0$.

If $\nu = \pm$, $[\sum_{i=L}^0 \alpha_i]^\mu \rightarrow [\sum_{i=M}^{L+1} \alpha_i]^\nu = [\sum_{i=M}^0 \alpha_i]^\mu$. If $\nu = +$ or $\nu = -$, the result depends on the form of α_L (w.l.o.g we may suppose that it is nonzero).

If $\alpha_L = \alpha'_L + \exp^L(\omega)\omega^p$, then the result is $[\sum_{i=M}^0 \alpha_i]^\mu$ as well. Otherwise, it is $[\sum_{i=M}^L \alpha_i]^\nu$.

This concludes the proof of the effective closure by \diamond .

B.4 Closure by \sup^+

We have to show that for $[\alpha]^\mu, [\beta]^\nu \in \Omega$, we can compute $\sup_i^+([\alpha]^\mu)^{(i)} \diamond [\beta]^\nu$, and the result is in Ω .

Case $\alpha = \exp^N(\omega)\omega^p + \alpha'$. If $[\alpha]^\mu = [\exp^N(\omega)\omega^p]^+$, then $([\alpha]^\mu)^{(i)} = [\alpha]^\mu$ and $\sup_i^+([\alpha]^\mu)^{(i)} \diamond [\beta]^\nu = \sup_i^+[\alpha]^\mu \diamond [\beta]^\nu = [1]^+ \rightarrow [\alpha]^\mu \diamond [\beta]^\nu$. We conclude by previously proved closure properties.

Otherwise, $[\alpha]^\mu \geq [\exp^N(\omega)\omega^p]^-$. But then, by the lemmas of the previous section, it follows easily that $([\alpha]^\mu)^{(i)} \leq ([\exp^N(\omega)\omega^p]^-)^{(i+1)} = [\exp^N(\omega)\omega^p(i+1)]^-$. Thus,

$$\sup_i^+([\alpha]^\mu)^{(i)} \diamond [\beta]^\nu = \sup_i^+[\exp^N(\omega)\omega^p i]^- \diamond [\beta]^\nu = \sup_i^+[\exp^N(\omega)\omega^p i]^+ \diamond [\beta]^\nu.$$

Let $\sum_{n=M}^L \beta_n$ be the usual presentation of β with $\beta_L > 0$. There are only three cases to consider, all the others being verified just by combining those three.

1. If $L > N$, $[\exp^N(\omega)\omega^p i]^+ \diamond [\beta]^\nu = [\beta]^\nu$ and we conclude like in the first case.
2. If $M < N$, $[\exp^N(\omega)\omega^p i]^+ \diamond [\beta]^\nu = [\exp^N(\omega)\omega^p i]^+$, and $\sup_i^+([\alpha]^\mu)^{(i)} \diamond [\beta]^\nu = [\exp^N(\omega)\omega^p i]^+$.
3. Suppose that $L = N$. Let $\beta_N = \exp^N(\omega)(\omega^{p+1}\eta + \omega^p j) + \beta'_N$ with $\eta < \omega^\omega$, $j < \omega$, and $\beta'_N < \exp^N(\omega)\omega^p$. Then, by applying the properties of the operation \diamond given by Lemma 11, we reason as follows.
If $\nu = +$, $\beta'_N = 0$, and $j = 0$, we get $[\exp^N(\omega)\omega^p i]^+ \diamond [\beta]^\nu = [\sum_{n=M}^{N+1} \beta_n + \exp^N(\omega)\eta\omega^p]^+$, and since i does not occur in the previous formula, we obtain that

$$\sup_i^+([\alpha]^\mu)^{(i)} \diamond [\beta]^+ = \left[\sum_{n=M}^{N+1} \beta_n + \exp^N(\omega)\omega^{p+1}\eta \right]^+.$$

Suppose $\nu \neq +$ or $\beta'_N > 0$. Under each of the two conditions we have

$$[\exp^N(\omega)\omega^p i]^+ \diamond [\beta]^\nu = \left[\sum_{n=M}^{N+1} \beta_n + \exp^N(\omega)(\omega^{p+1}\eta + \omega^p(i+j)) \right]^+$$

and so

$$\sup_i^+([\alpha]^\mu)^{(i)} \diamond [\beta]^\nu = \left[\sum_{n=M}^{N+1} \beta_n + \exp^N(\omega)\omega^{p+1}(\eta + 1) \right]^+.$$

Finally, if $\nu = +$, $\beta'_N = 0$, and $j > 0$, it holds that

$$[\exp^N(\omega)\omega^p i]^+ \diamond [\beta]^\nu = \left[\sum_{n=M}^{N+1} \beta_n + \exp^N(\omega)(\omega^{p+1}\eta + \omega^p(i+j-1)) \right]^+$$

and we conclude like before.

Case $\alpha = \exp^N(p) + \alpha'$ or $\alpha < \omega'$. If $\alpha = \exp^N(1)$ and $\mu \in \{+, -\}$, we get $([\alpha]^\mu)^{\langle i \rangle} = [\alpha]^\mu$, and the claim follows. Otherwise, $[\alpha] \geq_w [\exp^N(1)]^\pm$ like before, by the properties of \diamond , we have

$$\sup_i^+([\alpha]^\mu)^{\langle i \rangle} \diamond [\beta]^\nu = \sup_i^+([\exp^N(p)]^\mu)^{\langle i \rangle} \diamond [\beta]^\nu = \sup_i^+([\exp^N(i)]^\mu) \diamond [\beta]^\nu.$$

Again, analyzing all possible forms of β , we obtain an explicit formula in each case.